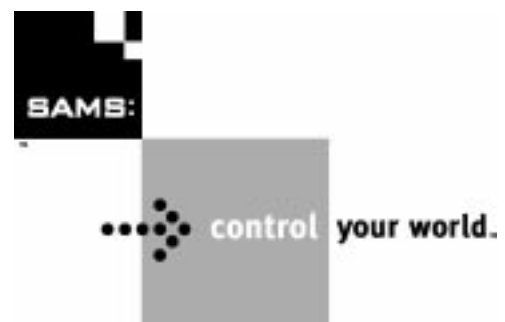




VOS3 Edition

Release 9.1
November 1998

User's Guide



© Copyright 1990-1998 Sterling Software, Incorporated
All Rights Reserved

Sterling Software, Inc.
Storage Management Division
625 East Carnegie Drive, Suite 100
San Bernardino, California 92408-3510

SMDS300P003-91

Proprietary/Trade Secret Notice

SAMS:Disk is undergoing constant revision, refinement, and expansion to include additional features. The current documentation is believed to be accurate and reflects the current version of the software. Important information that was not available when this book was published is contained in the Cover Letter that accompanied the distribution tape.

Warranty Disclaimer, Right to Revise

This documentation and the software it describes, constitute the proprietary, confidential, and valuable trade secret information of Sterling Software, Inc., and is licensed either "AS IS" or with a limited warranty, as set forth in the applicable license agreement. NO OTHER WARRANTY IS EXPRESSED AND NONE SHALL BE IMPLIED, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR USE OR FOR A PARTICULAR PURPOSE.

Sterling Software Inc., reserves the right to revise this publication from time to time and to make changes in the content hereof without the obligation to notify any person or entity of such revisions or changes. No part of this Message Manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, for any purpose, without the express written permission of Sterling Software, Inc.

Use, duplication or disclosure by the Government is subject to restrictions as set forth in Sections 52.227-14 and 52.227-19 of the FAR Commercial Computer Software Restricted Rights Clause and/or the Rights in Technical Data and Computer Software Clause at DFAR 252.227.7013 as applicable.

Trademarks

SAMS:Disk[®] VOS3 Edition is a trademark of Sterling Software, Inc., and has been registered in the United States. Applications in other countries, pending. The SAMS logo and Control Your World are trademarks of Sterling Software, Inc. Applications for registrations are pending in the United States and other countries. Trademarks of other products mentioned in this User's Guide are held by the companies producing them.

Customer Support

For Users in the United States, contact:

*Sterling Software, Inc.
Storage Management Division
10811 International Drive
Rancho Cordova, CA 95670
(800) 889-0226*

For Users Outside the United States

For technical support please contact your local Sterling Software customer support representative.

Table of Contents

Chapter 1. General Information	1
Overview of SAMS:Disk	2
Learning to Use SAMS:Disk	2
Basic Conventions and Requirements	4
Terminology	4
Control Statements	4
Data Set Names and Patterns	5
Tailoring to User Needs	8
Date Formats	8
Year 2000 Considerations	10
Separator Pages	10
Messages	11
User Abend Codes	11
Storage (Memory) Requirements	11
JCL Requirements	11
Data Set Allocation	12
Special Considerations	13
Types of Data Sets Processed	13
Direct Access and Physical Sequential Data Sets	14
RESTORE/RECOVER and Compression Products	15
Abnormalities Detected	15
Password-Protected Data Sets	16
Multivolume Data Sets	17
6951 MTL Automated Cartridge System	17
User Data Set Integrity	18
General Restrictions	20
Summary of Changes in this Release	21
Major enhancements include:	21

Chapter 2. Basic System — DMS PROC 23

Data Storage Command Language	24
Condition Codes	25
Messages	26
Commands	26
SCAN and FIND Rules	28
SELECT and EXCLUDE Rules	31
SET Command	35
PASSWORD Command (For VSAM Only)	36
SCAN Commands	38
SCAN REALVOLS Command	38
SCAN CATALOGS Command	39
FIND Command	41
SELECT Command	44
DSCL SELECT CRITERIA Field Name Descriptions	50
DSCL Select Criteria Boolean (AND/OR) Logic	68
EXCLUDE Command	70
ACTION Commands	73
Complex DSCL Command Sequences	74

Chapter 3. Basic System — FMS PROC 77

Overview	78
Condition Codes	78
Commands	78
FMS Command Sequence	79
General Rules Pertaining to the FMS Command Structure	80
SET Command	81
SYNTAX	81
PASSWORD Command (For VSAM Only)	83
SYNTAX	83
SCAN Command	84
SYNTAX	84
SELECT Command	85
SYNTAX	85
Criteria Statement Operands	87
FMS Select Criteria Boolean (AND/OR) Logic	95
EXCLUDE Command	96
SYNTAX	96
ACTION Commands	98

Chapter 4. Reports	99
General Information	100
Performance notes:	101
REPORT Condition Codes	101
Restrictions	101
REPORT Commands and Parameters	102
REPORT Command Names	102
Report Parameters	106
VREPORT Command Names	107
REPORT Field Legend	109
CLDATA Description	111
CIA Report Description	116
CIA Report KSDS Catalog Information Section	116
Index Component Catalog Information	118
Description of Sequence Set Graphic Headings	120
Unreferenceable Data Control Intervals	122
CIA Report Data Component Graphic Analysis	123
Control Information Overhead	125
Description of Data Component Graphic Headings	125
CIA Report KSDS Data Component	126
Usable Free Bytes	127
Estimated New Record Additions	127
Estimated Dead Space	128
CIA Report ESDS Catalog Information Section	132
CIA Report ESDS Graphic Analysis	132
CIA Report ESDS Data Component Summary	133
CIA Report RRDS Catalog Information Section	133
CIA Report RRDS Graphic Analysis	133
CIA Report RRDS Data Component Summary	134
Analyzing the CIA Report	135
Analyzing RRDS Relative Record Data Sets	135
Analyzing ESDS Entry-Sequenced Data Sets	135
Analyzing KSDS Key-Sequenced Data Sets	136
Utilization By Date and Time Report	138
SMS Report JCL	138
Select Command and Parameters	138
Sample DSCL REPORTS	140
ALLOCS	140
ATTRBS	140
CIARPT - KSDS Cluster	141

CIARPT - ESDS Cluster	142
CIARPT - RRDS Cluster	148
CLDATA	149
DISTR	150
DSINDEX	151
DSUTIL	152
FREESP	153
LAYOUT	154
MAPPER	155
MEMBER	156
MVDICT	157
PDSTAT	158
POCCHH	159
PODUMP	159
POTTRD	160
POXREF	160
PO1REF	161
PO2REF	162
SVDICT	163

Chapter 5. BACKUP/ARCHIVE 165

General Information	166
Terminology	166
BACKUP and ARCHIVE Considerations	168
Change Bit	168
Midnight “Rapid Aging” Problems	168
Exempting Data Sets	168
Exempting Restored Data Sets	168
Backup of Volume VTOCS	169
Task Performance	169
Task Restart Considerations	170
ARCHIVE Report Sequencing Options	170
Data Set Compression and Reorganization	170
Non-VSAM Considerations	171
Backup of SYSCTLG	171
PDS Anomalies	171
Model DSCB Processing	171
VSAM Considerations	172
Verify of VSAM Clusters	172
Backing Up or Archiving Empty Clusters	172

Processing Clusters with Alternate Indexes	173
Deleting Unexpired Data Sets	173
Processing Clusters in Use by Other Tasks	173
Recataloging for VSAM Auto-Restore	174
VSAM Backup and Archive Restrictions	174
Access Methods for VSAM Clusters	174
Alternate Index Support	175
DMF/EF Type-2 Catalog Backup and Recovery	176
Processing Clusters that have Non-Standard Formats	178
Incremental Backups	180
General Information	180
Implementing Incremental Backup	180
BACKUP Command	185
Overview	185
Restrictions	185
Syntax	186
BACKUP Command Examples	188
ARCHIVE Command	190
Overview	190
Restrictions	190
Syntax	191
ARCHIVE Command Examples	193
DELETE Command	195
DELETE Command Examples	196
EXPIRE Command	197
EXPIRE Command Examples	197
Sample Archive/Backup/Delete Report	198
EXPLICIT ARCHIVE	199
General Information	199
Restrictions	201
Parameters	202
Explicit ARCHIVE MODESET Command Parameters	202
Catalog Actions	204
Delete Actions	204
Reset Change Bit Actions	205
Processing Queued ARCHIVE Requests	205
Erasing of Deferred ARCHIVE Requests	206

Chapter 6. RESTORE/RECOVER 207

General Information	208
Multiple Files Data Set for Restore.	208
Restore Condition Codes.	208
Non-VSAM Restore Volume Selection	209
Multi-volume Restore	209
Deferred Restores (Queued Restore Requests)	210
Restore of SYSCTLGS	210
Restore Grace Periods	210
Restore to Preallocated Data Sets	211
PDS Member Restore	211
Restore Space Allocation	211
Restore Considerations for GDG Data Sets	212
Restore and Recover of VSAM Clusters	213
Restrictions in Restore and Recover of VSAM Clusters	213
Special Processing Options	214
Secondary Extent Consolidation (Defragmentation of VSAM Clusters)	214
Default Catalog Processing	214
Speed Option	214
Changing Data Sets Attributes	215
Empty Clusters	215
Deleting Unexpired Data Sets	215
Last Use Date Maintenance	215
Overriding Erase Option When Deleting	216
Using NOSCRATCH Option When Deleting	216
Processing Clusters in Use by Other Tasks	216
Restoring to Predefined and Reusable Clusters	217
Optional Cluster Data Report at Restore Time	217
Determination of Data and Index Component New Names	217
Determination of Alternate Index and Path New Names	218
Restoring Alternate Indexes	218
Volume Selection for Alternate Indexes	219
Restoring Alternate Indexes which were Archived Separately	219
Volume Selection for Multi-volume Clusters	220
DMF/EF Type-2 Catalog Recovery	221
Files Data Set Updates during RESTORE/RECOVER	222
Use of Primary or Copy Archive Volumes	222
Restrictions	222

RESTORE JCL	223
RESTORE Password Command	224
RESTORE Command and Parameters	225
Parameters Honored Without the Defer Parameter	227
Parameters for Non-VSAM Data Sets	227
Parameters for Non-VSAM Without DEFER	227
Parameter for VSAM Clusters	228
Parameters for VSAM Without DEFER	228
Parameters for VSAM AIXS Without DEFER	229
Parameters for VSAM Clusters Without DEFER	230
Parameters for VSAM Data Components Without DEFER ..	234
Parameters for VSAM Index Components Without DEFER ..	237
JCL for Processing Queued Requests	239
Tape Pull List for Queued Requests	240
Listing of Deferred Requests	240
Erase a Deferred Request from the Queue	241
Implicit Recovery and Volume Defragmentation	242
RECOVER Command	244
General Information	244
SYNTAX	245
Examples	247
Volume Defragmentation	248
Sample Reports	249
RESTORE	249
RECOVER	249
 Chapter 7. VBACKUP/VRECOVER	 251
General Information	252
VBACKUP Function	253
Track Overflow Data Sets	253
VBACKUP Security Considerations	254
Data Integrity Considerations	254
ARCHIVE Management Considerations	254
VBACKUP Command and Parameters	255
VRECOVER Function	257
Security Considerations	257
VRECOVER Command and Parameters	258
SAMPLE REPORT	261

Chapter 8. XCOPY	263
Overview	264
Limitations	264
Basic Operation	265
Automatic Selection	265
Secondary Files Data Set	265
Expiration Date Support	265
Condition Codes	266
Implementation	267
Create a Secondary Files Data Set	267
Identify Data Sets for Disaster Recovery	267
Determine Expiration Criteria	267
Running XCOPY	268
System Parameter Considerations	268
Simulate Mode Considerations	268
XCOPY Commands and Parameters	270
SET MODE= Command	270
Parameters	270
Syntax	270
Example	270
SELECT Command	271
Parameters	271
Syntax	271
Example	271
XCOPY Command	271
Parameters	271
Syntax	274
Example	274
Sample JCL	275
Sample OUTPUT	276
User Exits	277
XCPARCEXmmmmmmmmmm	277
XCPDSNEXmmmmmmmmmm	277
Record Formats	278

Chapter 9. Files Data Set Maintenance	279
General Information	280
Listing Archive Indexes.....	283
JCL - Index Listings	283
LISTV Command	284
LISTV Flags.....	284
LISTD Command	286
SYNTAX.....	286
LISTD Flags.....	291
LISTD Report Field Description.....	293
LISTF Command	294
SYNTAX.....	294
LISTF Report Field Descriptions	295
Sample LISTD, LISTF and LISTV Reports	295
IXMAINT Utility	296
General Information.....	296
Condition Code Settings	297
JCL - IXMAINT	297
Deleting Expired Data Sets	297
DSNDELETE Command and Parameters.....	298
Deleting Expired Volumes	299
VOLDELETE Command and Parameters	301
Purging Archive Data Sets/Volumes.....	302
PURGE Command and Parameters.....	302
Archive Index Rebuild Utility.....	304
JCL - Rebuild Index.....	305
REBUILD Command and Parameters	305
IXUPDATE Utility	306
JCL - Index Updates	306
Condition Codes.....	306
DELETE Data Set Index Command and Parameters	306
CHANGE Data Set Index Command and Parameters	307
RESET Command and Parameters	307
BUILDARC Command and Parameters.....	308
SYNTAX.....	308
Making Additional Copies of Archive Tapes.....	310

Chapter 10. MERGE	313
Overview	314
Specifying What to Merge	314
Specifying Merge Output Options	315
Implementation Concerns	316
Operation Concerns	316
TAPEPULL Processing	317
Abend/Restart Considerations	319
Simulate Mode Considerations	319
Processing Logic	321
JCL - MERGE Command	323
Condition Codes	324
MERGE Command and Parameters	325
SYNTAX	325
Sample MERGE Reports	329
 Chapter 11. MOVE/COPY	 331
General Information	332
ENQ/DEQ	332
Scratch/Catalog Defaults	332
Target Allocation	332
Excluding Data Set Names from Migration	333
Renaming Non-VSAM Data Sets	333
Renaming VSAM Data Sets	333
Messages	334
Restrictions	334
Moving Data to Different Device Types	335
Special Considerations	336
GDG Data Sets	336
Model DSCB processing	336
BDAM Data Sets	336
SYSCTLG Processing	337
Reblocking	339
Specifications	339
User Exits	341
MOVE Command and Parameters	342
COPY Command and Parameters	345
VSAM Cluster Considerations	348
Error Recovery	349

Sample Move/Copy Commands	350
Sample Move/Copy Report.	351
Chapter 12. Idle Space Release	353
General Information	354
Restrictions:	354
Messages	354
Condition Codes.	354
System Parameters.	355
RELEASE Command and Parameters	356
Examples	356
Sample Report	358
Chapter 13. PDS Compression	359
General Information	360
Restrictions:	360
Condition Codes.	361
JCL — PDS Compression.	362
SCAN Command and Parameters	362
COMPRESS Command and Parameters.	367
RELOAD Command	369
Sample PDS Compression Report.	370
Chapter 14. Sequential Migration to Tape	371
General Information	372
Installation Options	372
Criteria for Selection Data Sets	373
Scan Parameters.	373
Select Parameters.	373
Expiration Dates Assigned to Data Sets on Tape	374
Sequential Migration Versus Archival	374
Multivolume Tape Processing.	374
Compatibility With Tape Management Systems.	375
Creation of Sequential Migrate Tape First File.	375
Assignment of Creation Dates to Output Data Sets	376
Migrating Protected Data Sets.	376
Reports	377
Restrictions.	377
Condition Codes.	377

Sequential Migrate Commands	378
MODESET Command and Parameters	378
MIGRATE Command and Parameters.....	378
SCAN Command and Parameters.....	379
Migration Concerns	382
Catalog Concerns	382
Recatalog JCL - Recatalog to Duplicate Tape.....	383
RECATALOG Command—Parameter Descriptions.....	383
Sample MIGRATION Report.....	384

Chapter 15. DASD Billing 385

General Information	386
Elapsed Mode	386
Immediate Mode	387
Billing Rates	388
Computing the Billing Charge	388
Assign Volumes to Billing Categories	388
Define Billing Rates for Each Category	388
Example of Assigning Billing Rates.....	389
Billing Units System Parameters	389
DSBUNITSuuu - system parameter	389
Using Other Currencies.....	390
User Code/Account Codes	391
Assigning User Codes.....	391
Method 1: Displacement.....	391
Method 2: PARMLIB Member USERCODT	392
Method 3: Index Level (Node)	392
User Code Description Table	392
Total Charges	393
Exempting Data Sets.....	393
Deferred Write Precautions	394
DASD Billing Subfile Capacity.....	395
User Exits	396
Condition Codes	396
Accumulate JCL for Non-VSAM Data Sets	397
SELECT Command and Parameters.....	397
ACCUMULATE Command and Parameters	397
ACCUMULATE JCL for VSAM Clusters	400
DASD Billing Total JCL	401
Implementation - Suggested Run Procedures	402

Billing Processing Considerations	402
DASD Billing and Archival Functions	403
Increased Billing Buffers	403
Definitions of Fields in the DASD Billing Report	404
DASD Billing Samples	406
Sample by Device Type	406
Sample by User Code	407
Sample by Volume	407
 Chapter 16. TSS Command Processor	 409
General Information	410
DARCHIVE — Deferred Archive	411
Restrictions	411
DARCHIVE Command and Parameters	411
DRESTORE — Deferred Restore	413
DRESTORE Command and Parameters	413
RESTORE — Immediate (Dynamic) Restore	415
RESTORE Command and Parameters	415
LISTDMS — Listing the Archive Index	416
LISTDMS — Sample Output	417
LISTREQ — List Status of Deferred Requests	418
LISTREQ — SAMPLE OUTPUT	419
DERASE — Erase a Deferred Request	420
Restrictions	421
User Exits	422
 Chapter 17. Examples	 423
REPORT Examples	423
VREPORT Examples	425
ARCHIVE Examples	426
RESTORE Examples	428
Volume-Level BACKUP Examples	432
Volume-Level RECOVER Examples	433
Volume Defragmentation Examples	434
RECOVER Examples — FMS PROC	437
BACKUP Examples	443
Incremental BACKUP Examples	445
LISTD/LISTV Examples	446
IXMAINT Examples	447
MERGE Examples	448

Table of Contents

REBUILD Examples	450
IXUPDATE Examples	451
Idle Space Release Examples	452
Sequential Migrate Examples	453
PDS Compression Examples	454
Move/Copy Examples	455
DASD BILLING Examples	456
XCOPY Example	458
 Glossary	 459
 Index	 465

List of Figures

Figure 2-1. Sample DSCL REPORT Command	45
Figure 2-2. Sample DSCL Command Structure	74
Figure 4-1. DSCL Report Sample	102
Figure 4-2. Sequence Set Control Interval.	122
Figure 4-3. Sample Volume Allocation Summary Report	140
Figure 4-4. Sample Volume Attribute Summary Report.	140
Figure 4-5. Sample VSAM KSDS CIARPT	141
Figure 4-6. Sample VSAM ESDS CIARPT (1 of 6).	142
Figure 4-7. Sample VSAM RRDS CIARPT	148
Figure 4-8. Sample VSAM CLDATA Report.	149
Figure 4-9. Sample Volume Distribution DISTRs Report	150
Figure 4-10. Sample Data Set Index DSINDX Report	151
Figure 4-11. Sample Data Set Utilization DSUTIL Report.	152
Figure 4-12. Sample Volume Free Space FREESP Report.	153
Figure 4-13. Sample Volume Layout Report	154
Figure 4-14. Sample Volume Map Mapper Report	155
Figure 4-15. Sample PDS MEMBER Report	156
Figure 4-16. Sample MVDICT Report	157
Figure 4-17. Sample PDSTAT Report.	158
Figure 4-18. Sample POCCHH Report	159
Figure 4-19. Sample PODUMP Report.	159
Figure 4-20. Sample POTTRD Report	160
Figure 4-21. Sample POXREF Report	160
Figure 4-22. Sample PO1REF Report	161
Figure 4-23. Sample PO2REF Report	162
Figure 4-24. Sample SVDICT Report	163
Figure 5-1. Weekly Production Backup Schedule.	183
Figure 5-2. Sample BACKUP/ARCHIVE/DELETE Report	198
Figure 6-1. Sample RESTORE JCL	223
Figure 6-2. JCL for Tape Pull List.	240
Figure 6-3. Sample Restore Report	249
Figure 6-4. Sample Recover Report.	249
Figure 7-1. VBACKUP/VRECOVER Report	261
Figure 8-1. Sample JCL for XCOPY Command.	275
Figure 8-2. Sample OUTPUT From the XCOPY Command	276
Figure 9-1. Sample output of LISTD with FIELDS=	287
Figure 9-2. List Archive Index Report.	295
Figure 9-3. Sample JCL for Making Archive Tape Copies.	311

Figure 10-1. MERGE Command JCL.....	323
Figure 10-2. Sample MERGE Report.....	329
Figure 11-1. Moving a Data Set from 6587 to 6585.....	335
Figure 11-1. Sample MOVE Command JCL.....	350
Figure 11-2. Sample COPY Command JCL.....	350
Figure 11-3. Sample Volume Configuration Report.....	351
Figure 12-1. Sample RELEASE Command JCL.....	357
Figure 12-2. Idle Space Release Report.....	358
Figure 13-1. PDS Compress Report.....	370
Figure 14-1. DSNAMES Generated From MIGRATEC DD.....	376
Figure 14-1. RECATALOG JCL.....	383
Figure 14-2. Sample MIGRATION Report.....	384
Figure 15-1. Sample BILLING Command JCL.....	400
Figure 15-2. Sample ACCUMULATE Command JCL.....	400
Figure 15-3. Sample Billing Report - By Device.....	406
Figure 15-4. Sample Billing Report - By User Code.....	407
Figure 15-5. Sample Billing Report - By Volume.....	407

List of Tables

Table 1-1. Asterisk (*) Pattern in DSN=	5
Table 1-2. Question Mark (?) Pattern in DSN=	6
Table 1-3. Slash (/) Pattern in DSN=	6
Table 1-4. Exclamation Point (!) Pattern in DSN=	7
Table 1-5. Example of Finding the LLQ	7
Table 1-6. Patterns in VOL=	8
Table 1-7. User abend codes and common reasons	11
Table 2-1. DSCL Condition Codes and Descriptions	25
Table 2-2. DSCL Command Sequence	27
Table 2-3. Criteria Operators and their Meanings	47
Table 2-4. DSCL Selection Criteria Field Names	48
Table 2-5. CATSTAT Values and their Meanings	57
Table 2-6. DSORG Criteria Field Values	58
Table 2-7. GDGSTAT Criteria Field Values	59
Table 2-8. PASSTYPE Criteria Field Values	62
Table 2-9. PDSORG Criteria Field Values	63
Table 2-10. PSORG Criteria Field Values	64
Table 2-11. RECFM Criteria Field Values	64
Table 2-12. RECFM Criteria Field Values using Patterns	65
Table 2-13. VSAMORG Criteria Field Values	67
Table 3-1. FMS Condition Codes	78
Table 3-2. FMS Command Sequence	79
Table 3-3. Criteria Operators and their Meanings	87
Table 3-4. FMS Selection Criteria Field Names	88
Table 3-5. DSORG Criteria Field Value - FMS PROC	91
Table 4-1. DSCL Commands Available for Reports	99
Table 4-2. CLDOPT Values and Descriptions	103
Table 4-3. DSUTIL SYSPARM Displacement Values	104
Table 4-4. DSUTIL SYSPARM Displacement Variations	105
Table 4-5. Catalog Status Indicators	109
Table 4-6. Data Protection Indicators	110
Table 4-7. Use Statistical Group Definitions	113
Table 4-8. Password Group Protection Levels	114
Table 4-9. Password Replacement Values	114
Table 4-10. Attribute Detail Group Descriptions	115
Table 5-1. Explicit ARCHIVE Condition Codes	201
Table 6-1. Restore Condition Codes	208
Table 6-2. Tape Pull Condition Codes	240

Table 8-1. Condition Codes and Descriptions	266
Table 8-2. List of Cartridge Devices Simulated	269
Table 9-1. Files Data Set Maintenance Commands	282
Table 9-2. LISTD Condition Codes and Descriptions	283
Table 9-3. LISTV Flag Field - First Byte	284
Table 9-4. LISTV Flag Field - Second Byte	285
Table 9-5. Possible Values to LISTD FIELD= Parameter	287
Table 9-6. LISTD Flag Field - First Byte	291
Table 9-7. LISTD Flag Field - Second Byte	292
Table 9-8. IXMAINT Condition Codes and Descriptions	297
Table 9-9. IXUPDATE Condition Codes and Descriptions	306
Table 10-1. List of Cartridge Devices Simulated	320
Table 10-2. MERGE Condition Codes and Descriptions	324
Table 11-1. Possible REBLOCK Parameter Device Names	339
Table 12-1. Idle Space Release Common Sysparms	355
Table 13-1. PDS Compress Condition Codes	361
Table 13-2. PDS Compress SPACE Parameter Values	368
Table 14-1. Sequential Migrate Installation Sysparms	372
Table 14-2. Sequential Migrate Condition Codes	377
Table 15-1. DASD Billing Condition Codes	396
Table 16-1. LISTREQ Status Indicators and Descriptions	418

Chapter 1. General Information

SAMS:Disk is the automated data storage management system designed to assist the storage administrator in controlling and using data storage. Extensive reporting capabilities are provided, as well as functions to ensure sufficient free space, provide proper backup and recovery, enforce installation standards and perform device conversions.

SAMS:Disk gives you the flexibility to establish data management standards based on your installation's needs. It allows you to manage space differently, depending on the kind of data you have and how it is used. It lets you design and implement controls for DASD and tape usage throughout your storage hierarchy, and gives your data center's support staff and their end users an effective and convenient means to manage their data.

Overview of SAMS:Disk

The documentation provided with SAMS:Disk includes this User's Guide, which provides complete instructions for using all the SAMS:Disk functions, and examples of each. Also included is the Systems Guide, which provides comprehensive reference information for the data storage administrator who will be designing and customizing SAMS:Disk processing to meet the various needs of the data center. The Installation Guide provides instructions for installing and activating the SAMS:Disk system. A Message Manual documents all the messages issued by SAMS:Disk, along with a description of why the message was issued and instructions for corrective action if needed.

Learning to Use SAMS:Disk

This User's Guide contains all information necessary to run the SAMS:Disk system after it has been installed. It is organized into sections pertaining to the functions of the system. Each section describes what purpose the function serves, how to use it, and considerations for implementing it. Since this is a reference manual, it is not easy to learn how to use the system by reading the User's Guide from beginning to end. If you are new to SAMS:Disk, use the following list of items as a guide for getting to know the system.

1. Review the Table of Contents of the manual to get an idea of the functions the system provides and what information is supplied about them.
2. Continue reading this entire section of the manual. This will help you learn how the system operates, what its restrictions are, and how to code SAMS:Disk command statements.
3. Read the *Data Storage Command Language* section, beginning on page 23. This will help you learn how to write the SAMS:Disk command set used by most SAMS:Disk functions.
4. Review the *PARMLIB* section, beginning on page 437 of the *Systems Guide*. It describes the SAMS:Disk parmlib data set, which is used by all SAMS:Disk functions. It is very important to understand its purpose and how it is used.
5. At this point you should be prepared to learn about specific functions of the system. To do this, select a function that you are interested in and review the section for it. We suggest that you choose the REPORT function as the first to learn because it is the easiest to use and will provide a base of knowledge on how SAMS:Disk generally operates.
6. Once you have reviewed the section and feel you generally understand what it does and how to use it, go to the Examples section, beginning

on page 423 and find the examples pertaining to the function. Review each example.

7. Select one of the examples and tailor it for your installation. Key it into your system and execute it. Verify that it does what you expected. If not, consult the section on the function again.
8. Be sure that when running these test cases you specify the `SET MODE=SIMULATE` command or the `SIMULATE` command parameter for all functions that provide it. Once you have achieved the desired results with `SIMULATE`, remove the parameter (or comment it out), and run the function again, causing `SAMS:Disk` to actually perform the requested task.

Using these techniques, review each function you desire. Begin with simple cases, then add to them. They will help you to better understand the different options of each function and what they do. Keep track of your test output for future reference.

Basic Conventions and Requirements

Throughout the operation of SAMS:Disk, certain basic conventions and requirements are observed that help make processing consistent. These are summarized in the following pages.

Terminology

Several data storage management terms are commonly used from location to location, but the meanings given to the terms can vary widely. A Glossary is included in this *User's Guide*, beginning on page 459, to clarify both the common terms as well as those unique to SAMS:Disk.

Control Statements

All COMMANDS consist of a character string of ten characters or fewer. While a command does not have to begin in any particular position, nothing may precede it in a statement.

Following a command (and preceded by one or more spaces) is a series of PARAMETERS that may consist of a combination of keyword and simple entries. Parameters may be specified in any order for a given command.

Certain KEYWORD PARAMETERS may supply either a single value or a list of values contained within parentheses and separated by commas. The limit on the number of values in a list varies with each particular command. Common examples are lists of data set names and volumes.

COMMAND STATEMENTS may be continued to additional lines in two different ways. One technique permits termination of one statement line with a complete parameter and a trailing comma, with the remaining parameters starting in any position on the next line.

An alternate continuation technique requires coding of the statement being continued through column 71 and placement of an X in column 72 to designate that the statement is being continued. The first non-blank character in the following statement is appended to the character in column 71 of the previous statement.

Whenever any parameters are specified for a given command, the first parameter must begin in the statement containing the command. A statement with only a command is processed as having no parameters.

Parameter names may be abbreviated down to their first three characters, unless more characters would be needed to uniquely determine between similarly named parameters.

COMMENTS are permitted within the control statements. They are designated by an ASTERISK in column 1. They are printed out in the SAMS:Disk CMDPRINT exactly as they appear within the control stream.

Data Set Names and Patterns

The DSN= parameter is probably the most frequently used of all parameters, and the one where the greatest degree of flexibility is usually required. SAMS:Disk allows either full data set names or data set name patterns to be entered in the DSN= parameter or in any SAMS:Disk table that contains a data set name field. A pattern name consists of the usual alpha-numeric and national characters allowed in a data set name, but also the following:

1. An asterisk (*)
2. A question mark (?)
3. A slash (/)
4. An exclamation point (!)

The asterisk (*) may be used to represent any variable index level or simple name.

Table 1-1. Asterisk (*) Pattern in DSN=

Example	Result
DSN=*	will select all single-level data set names
DSN=*. *	will select all two-level data set names
DSN=A.*.SOURCE	will select all three-level names that have "A" as their first index and "SOURCE" as the simple name, but any second-level index
DSN=A*.SOURCE	will select all data sets having a first-level index beginning with the letter "A" followed by any other characters, and a second-level index equal to the value "SOURCE"

The question mark (?) may be used to represent any variable character within an index level or simple name. Multiple occurrences may be used within each level or simple name.

Table 1-2. Question Mark (?) Pattern in DSN=

Example	Result
DSN=?	will select all single-character data set names
DSN=A.TEST??	will select all two-level data set names with a first-level index of "A" and a simple name six characters in length, the first four of which must be "TEST"; for example, TEST01, TEST02, TEST99, TEST1A, TESTXX, etc.

The slash (/) may be used to represent any variable character from that position to the end of the name. The portion of the name that precedes the slash is referred to as a PREFIX name.

Table 1-3. Slash (/) Pattern in DSN=

Example	Result
DSN=A/	will select all data sets that begin with the letter "A"
DSN=A.TEST/	will select all data sets that begin with the character string "A.TEST", with any following characters
DSN=A.*.C?./	will select all data sets that begin with an index of "A", followed by any second-level index, a two-character third-level index starting with "C", and any following string

The exclamation point (!) may be used to represent any variable character up to the character string following the exclamation point; for example, the exclamation point defines the beginning of a character string (terminated by the next pattern character or the end of the data) that may be found anywhere within the name.

Table 1-4. Exclamation Point (!) Pattern in DSN=

Example	Result
DSN=!TEST	will select all data sets that contain "TEST" somewhere in the name
DSN=A?!.DEPT21	will select all data sets that have a two-character first-level index that starts with an "A" and contain "DEPT21" somewhere in the remainder of the name
DSN=!TEST!LOAD	will select all data sets that contain "TEST" somewhere in the name and "LOAD" somewhere following "TEST"

The exclamation point character may also be used to find a pattern occurring only at the end of a string. This can be accomplished by placing an exclamation point at the end of the desired pattern as well as at the beginning, as in the following example:

Table 1-5. Example of Finding the LLQ

Example	Result
DSN=!TEST!.LOAD!	will select all data sets that contain "TEST" somewhere in the name and which terminates with ".LOAD". Therefore, the data set name "LABS.TJP.TEST.LOAD" would match the pattern, but "LABS.TEST.LOADOLD" would not

Data Set Name Validation

SAMS:Disk commands that name a single specific data set to be processed are referred to as "explicit" commands. Those that cause VTOC and/or catalogs to be scanned are called "implicit" commands. By definition, the data set name entered on an explicit command may not contain any pattern characters. The specific name entered (original and perhaps new name) is validated according to the requirements as specified in the Hitachi JCL reference manual — rules for coding the data set name field. Invalid characters will cause the command to be rejected. Implicit commands do not have the data set name/pattern entries validated in this manner, since they process names that already exist in the VTOC or catalog.

Volume Prefixes

The VOL= parameter may specify either complete volume serial names or volume patterns; for example, a partial volume serial name with an appended slash (/) to signify any variable character from that position to the end of the field. SAMS:Disk tables that contain the volume field also allow volume prefixes unless specifically noted otherwise.

Table 1-6. Patterns in VOL=

Example	Result
VOL=(PACK01,WORK/)	will scan volume PACK01 and all volumes that begin with "WORK"
VOL=(/)	will scan all volumes

Tailoring to User Needs

Many system parameters exist that will cause slight variations from the default program execution. If a particular variation is desired, the appropriate sysparm is placed in the SYSPARMS member of the parmlib data set. Sysparms applicable to each SAMS:Disk function are cross-referenced and explained beginning on page 95 of the *Systems Guide*. A description is also given of how sysparms can be dynamically overridden in those special cases that warrant it.

For related information, please turn to the topic "*Activating SAMS:Disk Security Features and Interfaces*" on page 18 of the *Installation Guide*, to review how to protect these system parameters.

For larger scale variations, user exits are available. These are cross-referenced and explained beginning on page 195 in the *Systems Guide*.

Date Formats

You may enter dates as input parameters in any of the following formats (unless otherwise noted in the documentation of particular SAMS:Disk functions):

Gregorian Date Formats

ddmonyy	ddmonyyyy
mm/dd/yy	mm/dd/yyyy

where:

- dd is a calendar day of a month from 01 to 31
- mon is a calendar month such as JAN, FEB or MAR
- mm is a calendar month number from 01 to 12
- yy is a calendar year from 00 to 99 this year is stored as 1900 to 1999
- yyyy is a calendar year from 1900 to 2155

Julian Date Formats

```
yyddd  yyyyddd
yy.ddd  yyyy.ddd
yy/ddd  yyyy/ddd
```

where: ddd is a day of a year from 001 to 366
 yy is a calendar year from 00 to 99 this year is stored as 1900 to 1999
 yyyy is a calendar year from 1900 to 2155

Julian dates 98000, 99000, 99365, 99366 and 99999 are also accepted in any of the above formats.

For example, you may enter the date July 19, 1997 as:

97200	1997200
97.200	1997.200
97/200	1997/200
19JUL97	19JUL1997
07/19/97	07/19/1997

Also, as a DSCL Select or Exclude criteria, you may specify the date in the form TODAY, TODAY-n or TODAY+n, where “n” is a number of days in the past or future. For example:

- To select data sets created today you may enter SELECT CRITERIA=(CREDIT,EQ,TODAY)
- To select data sets modified within the last 5 days you may enter SELECT CRITERIA=(MODDT,GE,TODAY-5)
- To select data sets not used within the last 30 days you may enter SELECT CRITERIA=(USEDT,LT,TODAY-30)
- To select data sets expiring within the next week you may enter SELECT CRITERIA=(EXPDT,NE,0,AND,EXPDT,LT,TODAY+7).

SAMS:Disk treats Julian dates “99365”, “99366” and “99999” as “never scratch” dates. That is, data sets whose ARCHVOLS or DSNINDEX records use these dates as expiration dates will not be deleted automatically as expired by the SAMS:Disk function IXMAINT. In addition, if sysparm UNCATDEL defaults to a value of N, SAMS:Disk treats DSNINDEX records with an expiration date of 99000 as “never scratch.”

If you specify sysparm UNCATDEL with a value of Y, SAMS:Disk treats Julian date “99000” as a “catalog control date.” That is, data sets whose DSNINDEX record has an expiration date of 99000 will be deleted automatically by the SAMS:Disk function IXMAINT when the data set becomes uncataloged.

Year 2000 Considerations

The following information applies to JCL, System Parameters, or Command parameters involving EXPDT and RETPD values.

- If you explicitly specify EXPDT=99365 for an ARCHVOL or DSNINDEX record, then the data set is kept until explicitly deleted even after December 31, 1999.

For example:

```
SCAN REALVOLS
SELECT DSNAMES=PROD. /
BACKUP EXPDT=99365
//SYSPARMS DD *
DYNEXPDT=99365
```

The ARCHVOLS and DSNINDEX records created by this sample BACKUP job will have their respective EXPDT values set to 99365. These records will reside in the files data set even after December 31, 1999.

- If you specify RETPD=nnnn and the retention period calculates to an expiration date of 99365, the actual expiration date set is "bumped" to 2000.001 to prevent inadvertently assigning a "never-expire" value.

For example:

```
//jobcard
//BACKUP EXEC DMS
//ARCHIVE0 DD LABEL=RETPD=nnnn
SELECT DSNAMES=PROD. /
BACKUP RETPD=nnnn
```

The ARCHVOLS and DSNINDEX records created by this sample BACKUP job will have their respective RETPD values set to "TODAY+nnnn". If "TODAY+nnnn" calculates to 99365, the actual expiration date is "bumped" to 2000.001.

- If you specify RETPD=nnnn and the calculation extends beyond 2155.365 then the EXPDT is set to 2155.365.

CAUTION: RETPD values calculating to 98nnn and 99nnn may cause problems in certain situations. For example, TMS considers tapes valued at 99060 "scratch" after 60 days of non-use.

Separator Pages

The number of SAMS:Disk separator or header pages may be controlled by sysparm FLYCOUNT. From zero to nine pages may be produced. This sysparm

can be overridden in your *SYSPARMS* member in *PARMLIB*. For details, turn to page 96 in the *Systems Guide*.

Messages

SAMS:Disk MESSAGES all begin with a common format of MODULENM MSGN, where MODULENM indicates the name of the module issuing the message, and MSGN is the message number itself. This identifying information is then followed by explanatory text. Further explanations for each message are found in the *Message Manual*. A more detailed discussion of SAMS:Disk message handling can be found in the "Message Formatting" section, beginning on page 275 of the *Systems Guide*.

User Abend Codes

SAMS:Disk issues USER ABENDS when it is impossible to issue a diagnostic message, when additional information to diagnose a problem is needed, or when due to data integrity it is unsafe to continue.

Table 1-7. User abend codes and common reasons

USER ABENDS	COMMON REASONS
24	MSGPRINT DD statement missing
100	message will explain; dump attempted
200	message will explain; no dump attempted

Storage (Memory) Requirements

Buffers for SAMS:Disk I/O areas as well as memory tables for such items as member, alias and notelist entries are dynamically built. In some cases, SAMS:Disk does not create variable length memory tables above the 16MB line. Since these items vary for each data set and device type being processed, exact storage requirements cannot be given.

Regions sizes on supplied JCL procedures range from 1024 to 5120K to accommodate running SAMS:Disk with the maximum buffer space possible. The amount of region required below the 16MB line will vary depending upon the size of your catalogs, the number of your DASD volumes, etc.

JCL Requirements

JCL procedures are provided for most SAMS:Disk functions, documented beginning on page 11 in the *Installation Guide*. DD statements for any volume or data set that SAMS:Disk has been instructed to process are dynamically allocated as needed, freeing the user from any need to modify JCL.

Data Set Allocation

All SAMS:Disk function allocations use SVC 99 for dynamic allocation of non-VSAM data sets.

VSAM definitions are done using SVC 26, and are also candidates for being redirected by a pooling package.

Special Considerations

Types of Data Sets Processed

SAMS:Disk normally processes data by the movement and management of physical data blocks. In some cases, depending on actions requested by the user, SAMS:Disk may process at the logical record level. SAMS:Disk has its own “access method” (written at the channel program level) that does “full track reads and writes”, processing as much as one cylinder per I/O. In most SAMS:Disk functions, however, SAMS:Disk is not a “track-image” processing system.

Data is processed at the physical block level to allow SAMS:Disk to manage data without concern for source and target device dependencies, and to allow maintenance functions to be performed during data movement (for example, compress PDS data sets). (An exception to this type of processing is the SAMS:Disk volume-level processing for VBACKUP and VRECOVER, discussed below.)

For this reason, SAMS:Disk requires that all data sets to be processed be in a format that is standard to the VOS3 operating system environments. Processing is done for each data set based on the attributes that are carried for it in the format-1 DSCB of the volume VTOC. Attributes that are relied upon are DSORG and RECFM.

In general, if a data set’s internal record format(s) is not compatible with the standard meanings of the DCB DSORG attribute specified in the VTOC, SAMS:Disk may not be able to process it successfully. In cases where DCB attributes are not present, SAMS:Disk processes the data set with default attributes.

SAMS:Disk users should be aware of these kinds of data sets and investigate their compatibility before allowing processing against them. Data sets that are in question are generally special-purpose data sets. They may be part of an in-house generated system or used by a purchased product. In most cases they will be documented as being of a special format and maintainable only through utilities supplied with their associated system. These kinds of data sets usually constitute only a small fraction of the total DASD data sets in the user installation.

Track Image Processing for Volume-Level Functions

An exception to standard SAMS:Disk processing at the data set level explained above is the volume-level processing for the VBACKUP and VRECOVER functions. These functions back up and restore data at the volume level and use “track image” processing. SAMS:Disk does not examine separate data sets in these functions, and the data is restored exactly as it was backed up, but for an entire volume, not a single data set. Since no catalog action is taken during track image processing, you must verify that the catalog and recovered volume are in sync.

Direct Access and Physical Sequential Data Sets

As explained above, SAMS:Disk processing of data sets is based on the data set attributes. Sometimes these attributes conflict with the actual data or the attributes are incomplete. The following explains how SAMS:Disk will treat a variety of possible attributes during Archive, Restore, Move, and Copy processing of physical sequential and direct access data sets.

The attributes SAMS:Disk takes into consideration during processing include the DSORG, RECFM, LSTAR, BLKSIZE and KEYLEN. During processing, SAMS:Disk may also compress the data set, which will have an effect on whether or not TTR addressing is maintained.

The data set organization attribute (DSORG) and its types will be referred to as:

PS	physical sequential
DA	direct access
UNK	unknown

The record format attribute (RECFM) and its types will be referred to as:

F	fixed or fixed blocked
UV	undefined or variable
FBS	fixed block standard
UNK	unknown

The LSTAR is a pointer to the last record in the data set and is in TTR format. TTR format is a three-byte value. The first two bytes are the track number relative to the start of the data set, and the last byte is the record number relative to the start of the track. TTR addressing refers to a pointer for a data block.

Data set compression can occur when a data set is restored or moved to a DASD device with a different track length, if there is empty space on a track, or if there is an empty track in the middle of a data set. Each record is written on the next available space after the previous record. This can change the TTR address of the record and “compress” the data set. Under some conditions this is not desirable.

Data sets with a DSORG of DA and RECFM of UV require special handling. When a data set with DSORG of DA and RECFM of UV is archived by SAMS:Disk, an “ASDA” flag is set, indicating the data set is archived as a direct access data set. This flag is also set if an empty track is detected anywhere in the data set prior to the end-of-file (EOF). At restore time, if this flag is set, the TTR value for each record is maintained. That is, a record is written back in the same relative position from which it was read.

This also occurs when sysparm HONORTTR is specified with a value of Y, regardless of the RECFM of the PS, DA, or UNK data set.

Note: ESF Spool data sets must use HONORTTRY if the Move, Copy, Backup and Restore commands are to function properly.

Data sets are considered “empty” if their LSTAR value is zero. However, a data set whose DSORG is DA and RECFM is UV may have all of these attributes and not be empty. In this case, sysparm IGNLSTAR may be specified with a value of Y to cause SAMS:Disk to check bit 0 of the DSIND field in the format-1. For more information, see the sysparm description for *IGNLSTAR*, beginning on page 146 of the *Systems Guide*.

For data sets with a RECFM of FBS, the BLKSIZE is important. On any other PS or DA data sets, when the EOF is written, it is written on the next track after the last data track. An FBS data set must have the EOF on the same track if another record of BLKSIZE length can be written to that track.

SAMS:Disk handles any data set with a DSORG of UNK as a PS data set. Data sets with a RECFM of UNK are treated as though they have a RECFM of U. This corresponds to Hitachi’s handling of the same data sets. Data sets with a DSORG of UNK will have a DSORG of PS after SAMS:Disk processing.

Note: The Idle Space Release function will release space from UNK dsorg’s by default. You can instruct SAMS:Disk to bypass this type of DSORG by executing:

```
EXCLUDE CRI=(DSORG,EQ,UNKNOWN)
```

RESTORE/RECOVER and Compression Products

If your system has a data compression product installed that compresses VSAM data sets it may be necessary to exempt SAMS:Disk Restore and Recover processing from the data compression product’s processing.

SAMS:Disk reads VSAM data sets at the CI level when writing to the archive medium, for example, the data is read in a compressed format. However, during Restore processing, it writes to disk at the logical record level. This may cause certain data compression products to attempt to compress the data before writing it to disk. To prevent the data from being compressed twice, it is necessary to exclude SAMS:Disk from processing by the data compression package.

Abnormalities Detected

The SAMS:Disk functional capability detects and processes data sets with “external problems”, such as obsolete or unused data sets. The processing technique as described above also allows detection of “internal problems” or “unusual” conditions within the data sets, disk packs or catalogs themselves. Historically, this second type of detection has not been available at all (or only after it was far too late — an invalid data set shuts down a monthly job six hours into the run, and all the

track-image backups contain the same invalid track images). Consequently, users may be surprised by some of the diagnostic messages issued when SAMS:Disk is first implemented.

Whenever the integrity or recovery of the resource is not hindered by continued SAMS:Disk processing, a message is issued and execution proceeds. If the condition invalidates the above, messages are issued and the resource is bypassed if possible. Both of these cases provide the user with quick identification of problems, and hence the opportunity to make corrections in a more timely manner.

For example, if the directory of a PDS has been written over, default processing will issue messages to this effect and refuse to copy it to tape. Recovery of the valid backup copy of the data set can now be accomplished prior to its actual need, an ability that is usually greatly appreciated and much preferred over discovering too late that the backup copies are just as bad as the online data.

For related information, turn to the topic *"Processing PDSs that Contain Anomalies"* on page 69 of the *Systems Guide*.

Following are some of the types of diagnostics that can be expected:

- invalid PDS directories.
- wrong length records.
- multivolume data set CATALOG and VTOC disagreements.
- the existence of user labels on disk data sets.
- direct access data sets that are marked as sequential.

Password-Protected Data Sets

SAMS:Disk will support the processing of password-protected data sets in all functions.

If the installation desires to allow SAMS:Disk to process protected data sets, specify sysparm PASSWORD with a value of Y.

Normally, password-protected data sets cannot be renamed by SAMS:Disk. To override this feature, use the PASSNEWN sysparm as described on page 153 in the *Systems Guide*.

When dealing with password-protected data sets, SAMS:Disk does not take any action with the system password data set. It is the user's responsibility to take all necessary action to keep passwords correct and up to date.

Multivolume Data Sets

SAMS:Disk backup and archive facilities for VSAM, sequential (PS) and direct (DA) data sets include the ability to process data sets that span up to 20 DASD volumes. SAMS:Disk reports handle data sets on any number of volumes. All other functions that encounter multivolume data sets will print an error indicating the data set has been bypassed, and then go on to the next data set.

Processing of multivolume data sets requires that the data set be cataloged. The volume serial numbers found in the catalog will be used to determine the location of all the parts of the data set. An improperly cataloged data set may cause loss of data.

In implicit mode, SAMS:Disk will select multivolume VSAM for processing whenever any component is found, even if the data and index components reside on different volumes. For multivolume non-VSAM, selection is made only when the first volume of the data set is selected for processing.

For non-VSAM data sets, you may specify if multivolume data sets are to be processed by including sysparm ARCMULTI for explicit archives, or RETMULTI for implicit archive. The default specification is N, which causes SAMS:Disk to bypass archive or backup processing for all multivolume non-VSAM data sets.

SAMS:Disk will restore VSAM data sets to multiple volumes as needed, and non-VSAM data sets to either the first volume or to the original volume allocation depending on the value of sysparm PRIALLOC. If there are problems during restore, preallocating the data set usually solves the problem. For detail information about preallocation, review the sysparm description for *PREALLOC* on page 156 (non-VSAM), or *VSPREDEF* on page 191 (VSAM), both in the *Systems Guide*. If the restore does not provide the desired results, the *EXTRACT* utility, documented on page 291 in the *Systems Guide*, can be used to restore the data set to one or multiple volumes.

6951 MTL Automated Cartridge System

SAMS:Disk functions that use tape or cartridge devices are compatible with the 6951 MTL Automated Cartridge System, also referred to as “Nearline” storage. Devices under its control appear to SAMS:Disk as any other 6485-type devices. No changes to SAMS:Disk procedures are necessary to use them.

Installations that have the 6951 or similar device installed will find the benefits of an automated tape cartridge handling system particularly valuable for such SAMS:Disk functions as Archive and Backup, auto-restore, Sequential Migrate to Tape, and archive management functions.

User Data Set Integrity

To protect the integrity of your user data sets, before SAMS:Disk processes any data set, SAMS:Disk verifies whether data sets are in use by other tasks. This is done by the system enqueue service. This service works well in both a single CPU environment and a multiple CPU environment. However, the multi-CPU performs better if you have a cross-system enqueue package installed.

In multiple CPU environments with cross-system enqueue packages, you must ensure that all enqueues are propagated across all CPUs. For auto-restores in particular, enqueues for DMSAUT0, DMSAUT1, DMSAUT2, DMSAUT3, and DMSAUT4 must be propagated accurately. These enqueues ensure that data integrity is maintained.

If you have multiple CPUs but do not have a cross-system enqueue package installed, you must run SAMS:Disk at times when the other CPU does not need access to those data sets, or provide integrity through other means.

Enqueues using the normal system enqueue service are not always propagated across all systems in a JSS4 environment. SAMS:Disk solves this exposure by enqueueing on data sets using SVC 99. Enqueues resulting from a SVC 99 request are propagated in a JSS4 environment and provide a more reliable means of protecting data set integrity.

Note: JSS4 customers must specify a "Y" for sysparm ENQGDGBY because the SAMS:Disk JSS4 enqueue allocation technique automatically causes an enqueue of the GDG base. For details, please turn to the sysparm description for *ENQGDGBY* on page 141 in the *Systems Guide*.

For any function (such as Archive, Restore, Release and Move) that requires writing to, deleting or a catalog change to a data set, SAMS:Disk will attempt to get an exclusive enqueue on the data set. If any other task has a shared (DISP=SHR) or exclusive (DISP=NEW, OLD or MOD), SAMS:Disk assumes that someone else is using the data set, and SAMS:Disk will bypass its use of the data set.

For any function (such as Backup and Copy) that requires reading a data set, SAMS:Disk will attempt to get a shared enqueue on the data set. If other tasks have shared (DISP=SHR) enqueues, SAMS:Disk will process the data set. If any other task has an exclusive (DISP=NEW, OLD or MOD) enqueue, SAMS:Disk assumes that someone else is updating the data set, and SAMS:Disk will bypass its use of the data set.

SAMS:Disk Backup and Copy enqueueing techniques follow the standard linkage editor enqueue technique for load libraries.

The PDS Compress function provides an additional facility to protect user data sets in the shared DASD environment. It allows a command parameter (RESERVE) to be specified that causes SAMS:Disk to lock out all users from access of the volume on which the object data set resides. Lockout occurs during processing for each data set. When this occurs, all tasks in all other machines that need to access the volume

must wait for the compress to complete. Check with your installation's systems programming department before using this parameter, as it may severely degrade your machine throughput.

The VBACKUP and VRECOVER functions optionally issue a RESERVE and an ENQUEUE on the VTOC. This locks out all users on other machines from accessing the volume and it also prevents other users on the same machine from acquiring space on the volume. Because of reduced opportunity for a data integrity exposure, SAMS:Disk simulate mode and Report processing generally do not issue enqueues, and reports on data sets even if they are in use by other tasks.

Also see “*Processing Clusters In Use by Other Tasks*” on page 173 VSAM integrity concerns.

General Restrictions

The following is a list of restrictions you should consider when planning for SAMS:Disk. This list applies to the SAMS:Disk product in general. Any restrictions that pertain to specific functions are documented with each function separately. Although this list may seem large, the data sets that cannot be processed by SAMS:Disk in most installations constitute a very small percentage of the total data sets to be managed. This list is to help you avoid problems often caused when plans to process data do not include exception conditions.

- Data sets with the “SYS1” first-level identifier will never be scratched by SAMS:Disk unless SCRASY1 was specified to allow it. This is to protect system data sets from accidental loss.
- It should be noted, however, that SAMS:Disk does not do any special checking before taking catalog update actions for “SYS1” data sets.
- Multivolume VSAM, PS and DA data sets in the VOS3 environment can span up to 20 volumes for the archive and backup functions. Restores may also span up to 20 volumes.
- The following data set names are reserved for SAMS:Disk internal use and will not be processed by ARCHIVE and RESTORE functions:

VTOC . xxxxxxx
DMSOS . Vxxxxxxx

- The SAMS:Disk functions of Archive, PDS Compress, Move and Copy will not process data sets that have been selected implicitly if they are included in the APF list or linklist. To be processed, these data sets must be selected explicitly.
- User labels on disk or tape volumes are not supported. It should be noted that it is very uncommon for installations to implement user labels. User labels should not be confused with “standard labels” that are present on most tape and DASD volumes. Messages are issued to identify any data set containing user labels, and processing for the data set is bypassed.
- Data sets marked as unmovable are normally skipped. SAMS:Disk does not do absolute track allocation, nor does it use track-image processing, both of which are needed to ensure the integrity of such data sets. Sysparms are available in many cases, however, that will permit SAMS:Disk to copy (and perhaps relocate) them.
- SAMS:Disk does not support the PANEXEC program product data set load libraries. These data sets are formatted like a PDS but have no valid directory. SAMS:Disk cannot determine their internal format.

Summary of Changes in this Release

This release provides support for hardware and software changes from Hitachi and independent software vendors, along with many SAMS:Disk system enhancements. In addition, there are between 50 and 75 maintenance items included that provide corrections and improvements for the general operation of the system.

Major enhancements include:

RECOVER Command Significantly Enhanced

The RECOVER command has been significantly enhanced and is now executed under a DSCL-type language. Although the major benefits from this enhancement may surface during disaster recovery situations, you will notice significant improvements during day-to-day activities. For complete details on this enhancement, turn to the *FMS Overview* section on page 77.

Disaster Recovery Extract Utility (XCOPY)

XCOPY was designed to be an ideal compliment to a Data Centers disaster recovery procedure. While running XCOPY, you will benefit in lower CPU usage compared to that of your normal archive and backup production jobs. Your normal production window is not impacted, because the copy process uses an existing ARCHVOL and copies the data onto a Disaster Recovery ARCHVOL, eliminating all data set and volume selection overhead. These newly created ARCHVOLS are identical in file format as their archive and backup counterparts, they just reside in a separate files data set.

A set of flexible commands and parameters is provided, that allows Storage Administrators to copy only a subset of archive and backup data onto the new ARCHVOLS. Data sets can be selected by fully qualifying their data set names, or by selecting a special character and specifying a pattern of a data set name.

Finally, XCOPY has an internal mechanism that gives the Storage Administrator the ability to maintain different DSNINDEX and ARCHVOL expiration dates than that of their primary archive and backup copies.

For details on this enhancement and to find out how it operates, please turn to the "Overview" section in the XCOPY chapter, beginning on page 263.

LISTD Command Significantly Enhanced

This command is used to list out the contents of the DSNINDEX record(s). These are the records that are stored in the files data set whenever a backup or archive is issued for an online data set. Because there is so much information in these records (LRECL=256), it took 3 commands to list all of their contents. With this enhancement, you can now issue the LISTD command while selecting what information you wish to see.

The selection of information is controlled by a parameter on the LISTD command called **FIELDS**. For detailed information on how to use this parameter, turn to the *FIELDS=* parameter description on page 286.

New DSCL Parameters

The Data Storage Command Language (DSCL) has been enhanced to include three important functions that were previously only available under explicit archive. You can now add the following DSCL parameters to your BACKUP and/or ARCHIVE jobs:

1. **DEFER** — The presence of this parameter copies the backup or archive commands into the ARCHCMDS queue to be processed at a later time. Details on deferred backup can be found under the topic *"Deferred Requests"* on page 185. For deferred archive, details can be found under *"Deferred Requests"* on page 190.
2. **NEWNAME=** — Normal DSCL processing creates DSNINDEX records with names equal to that of the source data set names. With the presence of this parameter, you can now create DSNINDEX records under a different name. Details on *renaming backup DSNINDEX records* can be found on page 187, or for *archive records*, on page 192.
3. **MEMBER=** — The presence of this parameter allows you to back up or archive a PO data set at the member level. For use with the BACKUP command, details can be found under *"Member Level Backup"* on page 185. For the ARCHIVE command, details can be found under *"Member Level Archive"* on page 190.

Another addition to the DSCL command structure is the ability to back up or archive a data set under a different high level qualifier. This is accomplished by specifying the parameter NEWHLQ on the BACKUP or ARCHIVE command. Details on this parameter for the *BACKUP command* can be found on page 187, or for the *ARCHIVE command*, on page 192.

Application Level Backup and Recover

The FIND and RECOVER commands have been enhanced to include the **TABLE=** parameter. This enhancement allows you to use the same table in both BACKUP and RECOVER.

Details on this enhancement are in 2 separate locations. For BACKUP, review the *TABLE=* parameter description on page 43. For RECOVER, review the *TABLE=* parameter description on page 86.

Chapter 2. Basic System — DMS PROC

All of the commands below are executed by the DMS JCL procedure:

- ARCHIVE
- BACKUP
- COPY
- DELETE
- EXCLUDE
- EXPIRE
- FIND
- MOVE
- PASSWORD
- RELEASE
- REPORT
- VREPORT
- SCAN
- SELECT
- SET
- VBACKUP

In this release, Data Storage Control Language (DSCL) is used by most functions of SAMS:Disk. See "*DSCL Command Sequence*" on page 26 for more details. DSCL processes both non-VSAM data sets and VSAM clusters.

Data Storage Command Language

The objective of the SAMS:Disk Data Storage Command Language (DSCL) is to provide a rich, comprehensive palette of commands for efficient, common processing of mixtures of non-VSAM data sets and VSAM clusters.

Within a single execution you may report on, back up, archive, delete, expire, move, copy, and/or release idle space in non-VSAM data sets and VSAM clusters. You may do this based on scans of VTOCs of disk volumes, scans of DMF/EF Type-2, VSAM or explicitly named OS CVOL catalogs, or the results of a catalog superlocate.

You also have a rich set of criteria to select data sets for and exclude data sets from processing as desired. And a rich set of action parameters lets you control exactly what is done to these data sets.

SAMS:Disk preprocesses your command stream to find any “overlapping requests”, that is, those that request processing of the same volumes or catalogs. Such requests are grouped together to allow all needed actions to be performed during a single pass of volume VTOCs, catalogs, or catalog superlocates.

The functions Sequential Migrate to Tape, Merge, Xcopy, Restore, and PDS Compression continue to use their own unique command sets. Functions using DSCL are invoked by executing the JCL procedure named DMS. In its simplest form, its use would look like:

```
//jobname JOB (acct,info),...  
//STEP EXEC DMS  
DSCL commands
```

Note: The FMS proc, used for data set recovery, is very similar to the DMS proc and its DSCL command structure. Please turn to page 77 for details.

To avoid unexpected execution of partially bad command strings, the DMS proc takes no action until all commands have been successfully understood, and all sysparm overrides have been validated.

To allow for sorted reports, the DMS proc has separate process, sort and print steps.

Condition Codes

The DMS proc returns the following JCL condition codes:

Table 2-1. DSCL Condition Codes and Descriptions

Code	Meaning
0	All actions were processed successfully and informational messages issued. Check to see if any messages are issued and if further action is required. Review the generated reports to make sure that the desired data sets were processed as stated.
4	<p>Some minor problems were encountered in processing, but it was possible to continue. The results may not be exactly what you wanted, but there seemed to be no harm in continuing. Warning messages were issued. Review these messages and the generated reports to make sure that the desired data sets were processed as stated.</p> <p>Causes of this return code include selected data set in use by another task, insufficient access authority, I/O error, disagreements between the VTOC and/or catalog for one or more individual user data sets, data sets found for processing but no processing is done because they did not meet the stated selection criteria, data set not processed because it is cataloged to the SAMS:Disk pseudo volser, or the data set was not selected because it is restricted for the specified action command. The last case can occur when trying to select a user catalog with action commands other than 'BACKUP' and 'REPORT'.</p>
8	<p>A major problem was encountered in processing, resulting in SAMS:Disk not even attempting some processing. Error messages were issued.</p> <p>Causes of this return code include inability to mount a volume, inability to get the Format-4 DSCB for a volume, volume-oriented backup error, catalog scan errors, or no data sets could be found for processing.</p>
12	<p>The entire function, or a large portion of it, could not be processed. Error messages were issued.</p> <p>Causes of this return code include task abends, or data sets selected for processing but bypassed by the action being attempted.</p>
16	Commands were in error. Error messages were issued and no action was taken.

If these condition codes do not suit you, you may install a user exit to modify these condition codes based on message numbers that are issued or other criteria. For details, turn to the topic *"User-Specified Condition Codes"* on page 81 in the Sys-

tems Guide. Also see the sysparm description for *DSCLOWRC* on page 136 in the *Systems Guide*.

Messages

During DSCL SCAN REALVOLS and SCAN CATALOGS processing, SAMS:Disk may issue error messages when “broken” data sets are encountered. If you do not want these messages to be issued, set the sysparm DSCLMSG to N. This will suppress “broken” data set messages without affecting the job step return code. For more information, review the sysparm description for *DSCLMSG* on page 136 in the *Systems Guide*.

Commands

The DMS proc is controlled by commands provided to the SYSIN dd statement. The commands are of four main types:

1. **Environment commands.** Those commands that set up the environment in which SAMS:Disk will run: SET and PASSWORD.
2. **Candidate commands.** Those commands that find candidate data sets: SCAN and FIND.
3. **Selection commands.** Those commands that select or exclude data sets to be processed, based on your own specific criteria: SELECT and EXCLUDE.
4. **Action commands.** Commands that take action upon the selected resources: REPORT, VREPORT, BACKUP, ARCHIVE, DELETE, VBACKUP, RELEASE, MOVE, COPY, EXPIRE, and BILLING.

The minimum information required for you to execute DSCL is a SCAN or FIND command to name the volumes, catalogs or data set names to be processed, and an action command.

You may add other commands to gain additional selection or functionality.

DSCL Command Sequence

DSCL commands are listed below in their proper sequence. Each command is optional, except as described below.

Table 2-2. DSCL Command Sequence

Type	Name	Parameters
Environment	SET	MODE=SIMULATE,
Environment	PASSWORD (VSAM only)	CATALOGS=, MASTER=
Candidate	SCAN	REALVOLS, SHOWVOL
Candidate	FIND	DSNAMES=, CATALOGS=, PASSWORD=, TABLE=
Candidate	SCAN	CATALOGS=, EXCCATS=, BACKUP
Selection	SELECT	DSNAMES=, COMPONENTS, TABLE=, VOLUMES=, VTOCS, CATALOGS=, CRITERIA=
Selection	EXCLUDE	DSNAMES=, COMPONENTS, TABLE=, VOLUMES=, VTOCS, CATALOGS=, CRITERIA=
Action	REPORT	CLDATA, CLDOPT=, CIARPT, CIAOPT=, DSINDX, DSUTIL, MAPPER, MEMBER, MEMBERS=, MVDICT, PDSTAT, POCCHH, PODUMP, POTTRD, POXREF, PO1REF, PO2REF, SVDICT, COMPONENTS, CMDGENDSN=
Action	VREPORT	ALLOCS, ATTRB, DISTRs, FREESP, LAYOUT
Action	BACKUP	RETPD=, EXPDT=, ACCESS=, RPT=CIA, CIAOPT=, NORESET, DEFER, NEWNAME=, NEWHLQ=, MEMBER=
Action	ARCHIVE	RETPD=, EXPDT=, ACCESS=, RPT=CIA, CIAOPT=, DISP=, DEFER, NEWNAME=, NEWHLQ=, MEMBER=
Action	DELETE	DISP=
Action	EXPIRE	DATE=
Action	VBACKUP	RETPD=, EXPDT=, DSNENQ=, RESERVE, RPT=, LSTAR

Type	Name	Parameters
Action	RELEASE	PCTUSED=, PCTSECONDARY=, CONVERT
Action	BILLING (VSAM only)	
Action	COPY	ABSOLUTE, BLKSIZE=, CATALOGALL, CONTIG, CREATE, CYL, NEWNAME=, NEWVOLUMES, PREALLOC, REDUCE, TOVOL=, TRACK, VATHRESH=
Action	MOVE	ABSOLUTE, BLKSIZE=, CATALOGALL, CONTIG, CREATE, CYL, FILL, NEWNAME=, NEWVOLUMES, PREALLOC, REDUCE, TOVOL=, TRACK, VATHRESH=,

SET and PASSWORD commands are optional. If you do specify a PASSWORD command, it applies to processing of VSAM password-protected clusters only.

Although the SCAN and FIND commands are each optional individually, you must specify at least one SCAN or FIND command before any action command.

SELECT commands are mandatory, unless you have sysparm SELECTRE specified with a non-default value of N. EXCLUDE commands are optional.

Although action commands are each optional individually, you must specify at least one action command for each SCAN/FIND group (defined later).

Since the purpose of each command is largely self-evident, general rules and guidelines for using these commands are described immediately following. Detailed explanations for most DSCL commands and parameters are presented later in this section. Detailed explanations for each of the action commands and parameters are presented in later sections in this manual.

SCAN and FIND Rules

You may specify any number of SCAN or FIND commands in a command stream. Multiple commands may also be intermixed for any given action command. The following rules are observed by SAMS:Disk in processing SCAN and FIND commands. The rules are complex enough that you may want to try them out with SET MODE=SIMULATE until you get the feel of them.

Note: SET MODE=SIMULATE is not necessary (and will be ignored) while executing the REPORT or VREPORT commands, because no "action" is taking place.

SCAN and FIND Rule #1

DSCL processes the specified requests in terms of logical blocks termed “SCAN/FIND groups”. A SCAN/FIND group begins with a SCAN or FIND command and ends when the next non-consecutive SCAN or FIND command is encountered, or at the end of the input command stream.

At least one ACTION command must be specified in every SCAN/FIND group, with the one exception that SCAN CATALOGS=,BACKUP does not require an action command.

In its simplest form, a valid SCAN/FIND group might look like:

```
SCAN REALVOLS
  SELECT VOL=/ ,DSN=/
  REPORT ALLOC
```

which would cause SAMS:Disk to scan all online volumes and produce a volume allocation summary report.

Another simple SCAN/FIND group:

```
FIND DSNAME=LABS.TJP. /
  REPORT MVDICT
```

would report on all data sets and clusters beginning with the prefix “LABS.TJP.”

While DSCL does not care about spacing, the indentation shown may help us in the examples that follow.

SCAN and FIND Rule #2

Consecutive SCAN and/or FIND commands apply to all subsequent selection and action commands, until another SCAN or FIND command is encountered.

For example, if you have multiple naming conventions, you could report on each of your data sets with:

```
FIND DSNAME=SBTJP. /
FIND DSNAME=LABS.TJP. /
  REPORT MVDICT
```

The report command would apply to each of the FIND commands.

To backup the data sets and clusters cataloged in three DMF/EF Type-2, VSAM or OS CVOL catalogs, you could use:

```
SCAN CATALOGS=VOL05.USERCAT
SCAN CATALOGS=VOL00.USERCAT
SCAN CATALOGS=VOL05.TESTCAT
  SELECT DSN=/
  BACKUP
```

You may also combine FIND and SCAN commands, although this would result in extra overhead if the data set name on the FIND command is defined in the object of the SCAN:

```
SCAN CATALOGS=DMF.TESTCAT
FIND DSNAME=LABS.*.TESTVSAM/
  SELECT DSN=/
    REPORT CLDATA
```

SCAN and FIND Rule #3

SELECT and/or EXCLUDE commands apply to all action commands that follow, until either a non-consecutive SELECT or EXCLUDE command is encountered, or until a SCAN or FIND command is encountered.

You may apply the preceding rule to one or more selection tests for multiple action commands. In the following example, the two catalogs named will be scanned and all data sets that meet either SELECT test will be both backed up and reported on:

```
SCAN CATALOGS=VOL05.USERCAT
SCAN CATALOGS=VOL00.USERCAT
  SELECT CRITERIA=(TRKS,GT,30),DSN=/
  SELECT CRITERIA=(ARCDT,LE,TODAY-30),DSN=/
    BACKUP
    REPORT MAPPER,CLDATA,SVDICT
```

In the following example, both catalogs will be scanned and those data sets that have not been modified within the last year will be archived. In addition, those data sets that use less than 25 percent of their allocated space will be reported on. Note that because the two SELECT commands are not consecutive, the first SELECT command has no effect on the ARCHIVE command. However, a data set could be selected by both action commands if it met both selection criteria:

```
SCAN CATALOGS=VOL05.USERCAT
SCAN CATALOGS=VOL00.USERCAT
  SELECT CRITERIA=(PCTUSED,LT,25),DSN=/
    REPORT MVDICT
  SELECT CRITERIA=(MODDT,LT,TODAY-36),DSN=/
    ARCHIVE
```

You may also combine FIND commands with SELECT and/or EXCLUDE commands:

```
FIND DSNAME=LABS.TJP.VSAMFIL
FIND DSNAME=LABS.DMS.VSAM/
  SELECT CRITERIA=(TRKS,GT,20),DSN=/
  EXCLUDE DSNAME=LABS.DMS.VSAMP/
    ARCHIVE
```

SCAN and FIND Rule #4

Any ACTION command applies to all SCAN and/or FIND commands in the current SCAN/FIND group.

This rule has already been demonstrated in previous examples. Whenever an action command is entered, the easiest way to determine which volume(s), catalog(s) and/or data set name(s) it will be processed against is to look up in the command stream and find the first SCAN or FIND command that precedes it. The real volumes, catalog, or data set name on that command, plus any others named on preceding consecutive SCAN and/or FIND command(s), will be affected by that ACTION command.

So far all of our examples have illustrated command input with only one SCAN/FIND group. Now let's look at an example with four SCAN/FIND groups. Remember that a SCAN/FIND group is detected by the presence of a non-consecutive SCAN or FIND command. In this example we will scan all real volumes and all DMF/EF Type-2 and VSAM (but not OS CVOL) catalogs and report on all data sets created before January 16, 1996. In addition, another catalog will be scanned to archive VSAM data sets that have not been modified in the last 30 days, but no report is desired for that catalog. The final catalog will be scanned and all data sets will be reported on.

```
SCAN REALVOLS
  SELECT CRITERIA=(CREDT,LT,1/16/96),DSN=/
  REPORT MVDICT
SCAN CATALOGS=/
  SELECT CRITERIA=(CREDT,LT,1/16/96),DSN=/
  REPORT MVDICT
SCAN CATALOGS=VOL00.TESTCAT
  SELECT CRITERIA=(MODDT,LT,TODAY-30),DSN=/
  ARCHIVE
SCAN CATALOGS=DMS.PRODUCT.CATALOG
  SELECT DSN=/
  REPORT CLDATA,CLDOPT=DIBEBBAC
```

SELECT and EXCLUDE Rules

You may specify any number of SELECT and EXCLUDE commands in a command stream. Multiple commands may also be intermixed for any given action command. The following rules are observed by SAMS:Disk in processing SELECT and EXCLUDE commands. Use SET MODE= SIMULATE until you understand the complexities of the rules.

SELECT and EXCLUDE Rule #1

For each SELECT or EXCLUDE command, all parameters specified must be true for the statement to be true. This is equivalent to AND selection based on the parameters on any individual SELECT or EXCLUDE commands.

For example,

```
SELECT VOLUMES=TSS/,CRITERIA=(DSORG,EQ,PS),DSN= /
```

would hold true only for sequential data sets on your TSS volumes, not for any other data set on your TSS volumes, nor for any sequential data sets on any other volume.

Note: SCAN CATALOGS or FIND commands compare the select volume list to the primary volume only for non-VSAM multivolume data sets.

SELECT and EXCLUDE Rule #2

If only SELECT commands apply to an action command, at least one SELECT statement must be true for the data set to be selected for SAMS:Disk processing. This is equivalent to ORing selection based on combining SELECT or EXCLUDE commands.

For example,

```
SCAN REALVOLS
  SELECT VOLUMES=TSS/ ,DSN=/
  SELECT VOLUMES=WORK/ ,DSN=/
  REPORT MVDICT
```

would report only on data sets on TSS and WORK volumes, not on any other volumes.

SELECT and EXCLUDE Rule #3

When multiple SELECT commands apply to an action command, if any SELECT statement is true for a particular data set, that data set will be selected for processing.

For example,

```
SCAN REALVOLS
  SELECT DSNAME=LABS.KSV. /
  SELECT VOLUMES=WORK/ ,DSN=/
  RELEASE PCTUSED=50
```

would release idle space from all VSAM and non-VSAM data sets that start with LABS.KSV, and would release idle space on all data sets (regardless of their data set names) on your WORK volumes. If you intended to release space on only LABS.KSV data sets on WORK volumes, you should instead specify both parameters on a single SELECT command, as in:

```
SCAN REALVOLS
  SELECT VOLUMES=WORK/ ,DSNAME=LABS.KSV. /
  RELEASE PCTUSED=50
```

SELECT and EXCLUDE Rule #4

VTOCs, and catalogs are not considered user data sets, and so are never processed unless specifically selected by SELECT VTOCS and SELECT CATALOGS=.

Consistent with rule 2 above, if you specify SELECT VTOCS or SELECT CATALOGS= and you wish to also process regular data sets as well, you must also supply a SELECT DSNAMES=/ command.

For example,

```
SCAN REALVOLS
  SELECT CATALOGS=/
  BACKUP
```

will back up only your catalogs, but,

```
SCAN REALVOLS
  SELECT CATALOGS=/
  SELECT VTOCS
  SELECT DSNAMES=/
  BACKUP
```

will back up your catalogs, VTOCs and data sets.

SELECT and EXCLUDE Rule #5

Consistent with rule 4 above, if you set sysparm SELECTRE to N and specify neither SELECT commands nor EXCLUDE commands, SAMS:Disk will process all user data sets, but neither VTOCS nor VTOC indexes or catalogs.

For example,

```
SCAN REALVOLS
  BACKUP
```

will back up your user data sets, but not your VTOCS nor VTOC indexes or catalogs. But,

```
SCAN REALVOLS
  SELECT CATALOGS=/
  SELECT VTOCS
  SELECT DSNAMES=/
  BACKUP
```

will back up your catalogs, VTOCs and data sets.

SELECT and EXCLUDE Rule #6

If you use sysparm SELECTRE with its default value of Y, you must enter at least one SELECT command in each SCAN/FIND group.

For example,

```
SCAN REALVOLS
    DELETE
```

would not be permitted.

SELECT and EXCLUDE Rule #7

If only EXCLUDE commands are specified for a given action command, if any EXCLUDE statement is met, the data set is bypassed. If no EXCLUDE commands test true, the data set will be processed.

For example,

```
SCAN REALVOLS
    EXCLUDE VOLUMES=PROD/
    EXCLUDE VOLUMES=SYS/
    SELECT DSN=/
    REPORT MVDICT
```

would not report on your PROD or SYS volumes, only on your other volumes.

SELECT and EXCLUDE Rule #8

If both SELECT and EXCLUDE commands are specified for a given action command, the following rules apply:

- If an EXCLUDE command tests true, the data set is bypassed whether a SELECT statement tests true or not.
- If no EXCLUDE tests are met, at least one SELECT command must still test true for the data set to be processed.

For example,

```
SCAN REALVOLS
    EXCLUDE VOLUMES=PROD/
    EXCLUDE VOLUMES=SYS/
    SELECT CRITERIA=(DSORG,EQ,VSAM),DSN=/
    REPORT CLDATA
```

would report on VSAM clusters on other than your PROD or SYS volumes. It would not report on non-VSAM data sets, nor would it report on any VSAM clusters on your PROD or SYS volumes.

SET Command

Use the SET command to define environment values that will be in effect for the duration of this execution only. If used, you may put the SET command anywhere in the command stream before the first action command. For clarity, we recommend that you specify this command before all other commands.

As with all SAMS:Disk commands, it can be commented out (nullified) by placing an asterisk “*” in column 1 of the line containing the command.

The syntax of the SET command, is as follows:

```
SET MODE=SIMULATE
```

MODE=

Specify SET MODE=SIMULATE or SET MODE=SIM to have SAMS:Disk simulate the execution.

To run in LIVE mode, either comment out the SET MODE= command from the input stream, or delete the line entirely.

In general, in simulate mode SAMS:Disk will produce the normal messages and reports as if processing had taken place, but will not alter any data sets. For example, reports do not alter any data sets, so reports run equally well under simulate mode or live mode.

PASSWORD Command (For VSAM Only)

The PASSWORD command is used to provide SAMS:Disk with any password necessary to carry out the functions requested in the command stream. Note that a catalog's master password is not required to scan a VSAM-password-protected catalog. Therefore, you need not specify any password to access a catalog. The proper password is required, however, to display passwords on the VSAM cluster data report, and also to archive or backup a cluster if it is password-protected. The command should precede the SCAN command that it is intended to cover. For clarity, we recommend that you specify this command just after the SET command, if any, and before all other commands.

The syntax of the PASSWORD command, is as follows:

```
PASSWORD MASTER=, CATALOGS=
```

MASTER=

This REQUIRED parameter is used to supply SAMS:Disk with the proper level of password it will need to carry out its requested functions. You may supply any of the following passwords:

- catalog master password
- cluster master password
- SAMS:Disk master password

The following rules govern when a password is required and the level of authority that is required:

- If the catalog is not password-protected, no password is required (even if clusters within the catalog were defined with passwords).
- If the REPORT function is being executed, no password is required, regardless of any catalog password protection. However, to display passwords on the cluster data report, the proper password must be specified. If either the catalog's master password or the SAMS:Disk master password is specified, all passwords will be displayed for all clusters that are protected in the catalog. If a cluster password is entered, all clusters will be listed on the report, but the passwords will be suppressed for those clusters that have passwords which do not match. If no password is supplied, no passwords will be displayed on the report.
- If a BACKUP command is executed, for those clusters that are selected and are password-protected, either the catalog's master password, the SAMS:Disk master password, or the cluster's read password must be specified. This same requirement applies for clusters that are being reported on in the cluster internal analysis report (CIA). In fact, you

must supply the control password for the data set (if one exists) because SAMS:Disk must read the cluster with control interval access to produce the report.

- If DMF/EF Type-2 catalogs are to be backed up (exported), the highest level password for each catalog should be specified, or the SAMS:Disk master password. If not properly specified, and unless “attempts=0” is in effect, the task will wait while the operator is prompted for the password.
- If an ARCHIVE command is executed, for those clusters that are selected and are password-protected, either the catalog’s master password, the SAMS:Disk master password, or the cluster’s master password must be specified.

CATALOGS=

Specify the catalog name for which the password value is to be used. If this parameter is omitted, SAMS:Disk will use the password specified on the MASTER= parameter for all catalogs that are specified on SCAN commands for which no explicit password is specified via the PASSWORD command. If more than one PASSWORD command is specified that does not name a catalog explicitly, the most recent PASSWORD specified without a catalog name will take precedence for any succeeding catalogs named on a SCAN command that had no PASSWORD explicitly named for them. (See the example on page 73 for further clarification of PASSWORD substitution rules.)

Because the password is specified on a separate command, it can be included from a different source than the remaining SAMS:Disk commands. Creating a protected data set containing the PASSWORD command and concatenating this data set ahead of the typical SAMS:Disk input set is one way to implement an easily maintained secured environment. When it becomes necessary to modify the password, simply replace the PASSWORD command also. This allows users to continue to process without knowledge of the required or changed security. SAMS:Disk will suppress the printing of all passwords on the CMDPRINT output.

SCAN Commands

The SCAN command is used to tell SAMS:Disk to scan the VTOCs of volumes, or to scan DMF/EF Type-2, VSAM or explicitly named OS CVOL catalogs. A SCAN or FIND command must precede any action commands. If any SELECT and/or EXCLUDE commands are used, they must follow the SCAN command(s) and precede the action command(s). If you request a scan of a protected catalog and you desire to supply a password to SAMS:Disk, a PASSWORD command should precede the SCAN CATALOGS= command.

On any given SCAN command, one and only one of the following parameters must be specified:

- REALVOLS
- CATALOGS=

These parameters tell SAMS:Disk what type of scan is to be performed. With REALVOLS, the volume VTOCs will be scanned, whereas the CATALOGS= parameter will cause DMF/EF Type-2, VSAM or explicitly named OS CVOL catalogs to be scanned.

SCAN REALVOLS Command

The syntax of the SCAN REALVOLS command, is as follows:

```
SCAN REALVOLS,SHOWVOL
```

REALVOLS

This parameter tells SAMS:Disk to scan real, online volumes. It will cause SAMS:Disk to process all real volumes in your installation that are currently available. The volumes to be scanned can be restricted by using the SELECT and/or EXCLUDE commands. More information regarding volume selection can be found in the SELECT and EXCLUDE command descriptions on pages 46 and 71 respectively.

SHOWVOL

Use this parameter to instruct SAMS:Disk to issue an explicit message (2860) to inform you when it starts processing a DASD volume (that is, begins to scan the volume's VTOC). The message helps relate any subsequent errors or warning conditions that SAMS:Disk may encounter to a specific DASD volume. The issuance of this message can be made automatic by specifying sysparm DLVOLMSG with a value of Y (the default value).

SCAN CATALOGS Command

The syntax of the SCAN CATALOGS command, is as follows:

```
SCAN CATALOGS= , EXCCATS= , BACKUP
```

CATALOGS=

Specify a catalog name (or catalog pattern) to be scanned. The catalog(s) may be either DMF/EF Type-2, VSAM or an explicitly named OS CVOL. If an explicit catalog name is specified (for example, PROJ99.USERCAT), only that one catalog will be processed. However, if a catalog pattern is specified (for example, SYS1/), all catalogs that match the pattern and are connected to the master catalog will be processed. All standard SAMS:Disk pattern matching capabilities are available (/ , ?, * and !).

Note that the pattern matching capability is not available in any operating environment that does not connect user catalogs to the master catalog. You may turn off this support by specifying sysparm VSCATPTN with a value of N, thereby forcing you to explicitly name a catalog. It should also be noted that the pattern matching capability does incur some additional overhead in that the master catalog must be searched in order to determine the user catalogs defined to the system. The master catalog will also be selected if it meets the specified pattern.

Normally, if a pattern name is specified that doesn't match any of the catalog names in your installation, SAMS:Disk will terminate all commands in the command stream. This can be overridden by specifying sysparm VSCATPTN with a value of C, such that only the one SCAN command will fail and all other commands will be executed.

EXCCATS=

This parameter should be specified only when a catalog pattern is specified on the CATALOGS= parameter. Its function is to allow the user to exclude up to ten catalog names and/or patterns that would otherwise be selected based on the pattern supplied on the CATALOGS= parameter. Refer to the example on page 73 for a further explanation.

BACKUP

This parameter causes a backup copy to be taken of each DMF/EF Type-2 catalog selected for processing by the SCAN command, with the retention period provided by the RETRETPD sysparm. SAMS:Disk uses the JSCVSUT EXPORT utility to create the catalog backup, but then copies its ALIAS entries from the master catalog as part of the backup as well.

Do not confuse this BACKUP parameter (on the SCAN command) with the BACKUP command itself. This BACKUP parameter causes DMF/EF Type-2 catalogs to be backed up. The BACKUP command causes VSAM or non-VSAM data sets or DMF/EF Type-2 catalogs to be backed up. If you would like to specify a particular retention period or expiration date for the backup of an DMF/EF Type-2

catalog, specify `SELECT CATALOGS=` and `BACKUP RETPD=` or `EXPDT=` under any `SCAN` command.

FIND Command

The FIND command is used to specify a data set name or pattern to be processed. The function of the FIND command is to improve the performance of SAMS:Disk when only one or a small subset of data sets are to be processed. Rather than scanning an entire catalog (as happens with the SCAN command), SAMS:Disk will issue one superlocate for the data set name or pattern specified. Catalog management returns all data set name(s) matching the data set name specified in one inquiry, thereby reducing catalog management overhead for the job.

SELECT and EXCLUDE commands may be used with the FIND command, although this would be useful only if a pattern is specified for the data set name. If any SELECT and/or EXCLUDE commands are used, they must follow the FIND command and precede the action command(s) for the specified data set name/pattern. You may use this command in the same job stream as the SCAN command, although it would be duplicating effort if the data set name specified on the FIND command would otherwise be processed as a result of the SCAN command.

When should you use the FIND command? Any time that you are interested in processing one cataloged data set or cluster, a few cataloged data sets or clusters, or a group of cataloged data sets or clusters that share the same high level qualifier or qualifiers.

As an example of how the FIND and SCAN commands can accomplish the same objective, look at the following:

```
SCAN REALVOLS
  SELECT DSNAMES=LABS.TJP. /
  ARCHIVE

SCAN CATALOGS=DMF.TESTCAT
  SELECT DSNAMES=LABS.TJP. /
  ARCHIVE

FIND DSNAMES=LABS.TJP. /
  ARCHIVE
```

In this instance the FIND command would probably be a better choice because much of the overhead associated with scanning VTOCs or a VSAM catalog is eliminated. However, a SCAN command with the SELECT command will accomplish the same objective.

The FIND command searches DMF/EF Type-2, VSAM and OS CVOL catalogs for entries. One disadvantage of using the FIND command is that catalog management will search only one catalog to find the data set name or pattern that you specify. In some situations, the catalog name should be specified on the FIND command so that catalog management can locate the data set. Two of these situations are: when the first-level node of the data set does not have an alias pointer in the master catalog, and when the data set is in the wrong catalog. Note that if the data sets you are

attempting to process are uncataloged, you must change your command and use `SCAN REALVOLS`.

The syntax of the `FIND` command, is as follows:

```
FIND DSNAME= , CATALOG= , PASSWORD= , TABLE=
```

DSNAME=

This parameter, **REQUIRED** only when the `TABLE=` parameter is not specified, is used to specify one explicit data set name or `SAMS:Disk` pattern. This parameter is mutually exclusive with the `TABLE=` parameter. All standard `SAMS:Disk` pattern matching capabilities are available (`/`, `?`, `*` and `!`).

Note: If a pattern is specified, the first node of the data set name must be specified without pattern characters. This is a requirement of catalog management, since the high-level node will be used to determine what catalog to search. For example, `“AB/”` is not the same as `“AB./”`, even if `“AB”` is a complete node. The former will require much more memory because the “end of the node” is not indicated, which limits the catalog locate function.

Users wanting to locate data sets with only a partial HLQ must use the `SELECT TABLE=` command on either a `SCAN REALVOLS`, or `SCAN CATALOG`.

CATALOG=

Use this **OPTIONAL** parameter to specify a catalog that catalog management should search to find the specified data set name or pattern. This parameter is required only if the first node of the data set name does not have an alias connector in the master catalog, or if the data set is in a catalog other than the one to which the alias points.

PASSWORD=

Use this **OPTIONAL** parameter when processing password-protected VSAM clusters to supply `SAMS:Disk` with the proper level password it will need to carry out its requested functions. You may supply any of the following passwords:

- catalog master password
- cluster master password
- `SAMS:Disk` master password

Refer to the command description for `PASSWORD` on page 35 to determine the level of password required for each function (`REPORT`, `ARCHIVE`, etc.). Note that this parameter must be specified even if the `PASSWORD` command is specified separately — the `PASSWORD` command is not used by the `FIND` command.

TABLE=

This parameter, REQUIRED only when the DSNAMES= parameter is not specified, is used to specify the name of a member in the parmlib data set containing the list (table) of fully qualified data set names or patterns. If a pattern is specified, the first node of the data set name must be specified without pattern characters. Finally, this parameter is mutually exclusive with the DSNAMES= parameter.

Note: Since SELECT and/or EXCLUDE commands can be combined with FIND TABLE=, data sets are not just automatically selected because they appear in this list. In the following example, the data set must both be listed in the parmlib member named "IMSDSNS" and must have been backed up within the last five days:

```
FIND TABLE=IMSDSNS
      SELECT CRITERIA=(ARCDT,GE,TODAY-5)
```

Values specified in the TABLE= member have the same limitations as those for the DSNAMES= parameter, described on page 42.

There is not an exact limit to the number of entries that can be entered into a TABLE= member. It all depends on the environment. The information for the table is compiled and stored in a Variable Length table in memory above the 16M line. So, the maximum number of table entries is directly related to the amount of memory available to DSCL.

The format of the TABLE= member begins in column 1 as follows:

```
'SYS/ '
'*. *.PRINT'
```

For detailed information on the TABLE= member, please turn to the "*Selection List*" section on page 455 located in the *Systems Guide*.

SELECT Command

The SELECT command allows a user to process only those data sets that satisfy the specified selection test(s). This command is required unless you have SAMS:Disk sysparm SELECTRE specified with a value of N.

If a SELECT statement is specified, in order for a data set to be selected, ALL tests specified on a single SELECT command must be met. If only one parameter is specified, only that single test must be met for selection to take place.

For example, if the DSNAMES= parameter or the VOLUMES= parameter, but not both, are specified on a single command, conceptually think “process only these data sets” or “process only these volumes”. If both the DSNAMES= and VOLUMES= parameters are specified, conceptually think “process only these data sets if defined on one of these volumes”.

If multiple SELECT commands are specified for any given action command, this has the effect of a logical “OR” between SELECT commands. That is, if any SELECT command is true, the data set is processed, regardless of whether other SELECT commands test true or not (see EXCLUDE exception below).

Note: If an EXCLUDE command immediately precedes or follows any SELECT command, special rules apply. Refer to the heading entitled “*SELECT and EXCLUDE Rules*” on page 31 for a further explanation.

```
SELECT DSNAMES= , COMPONENTS , TABLE= , VOLUMES= ,
      VTOCS , CATALOGS= , CRITERIA=
```

DSNAMES=

You may specify up to ten fully qualified names or SAMS:Disk patterns.

The reason for this option is that the SELECT and EXCLUDE commands control the data sets that are selected. If any SELECT command is specified without the DSNAMES= parameter, all data sets are selected. If multiple SELECT commands are entered and you want them to all process the same data sets, the data set list must be copied to each SELECT, as in the example below:

```
SCAN REALVOLS
  SELECT DSN=(SYS1/ , SYS2/ ) , CRITERIA=(TRKS , GT , 200)
  SELECT DSN=(SYS1/ , SYS2/ ) , CRITERIA=(EXTENTS , GT , 3)
  REPORT SVDICT , MVDICT
```

Because a SELECT DSNAMES= statement is already very restrictive, you may not specify the VTOCS, nor CATALOGS= parameters on the same SELECT statement. You may, however, add the COMPONENTS, TABLE=, VOL= or CRI= parameters.

For improved performance for VSAM while using the REPORT command, see the COMPONENTS parameter description on page 44.

COMPONENTS

Under “SCAN REALVOLS”, the COMPONENTS parameter in Reports will modify the action of SELECT DSNAMES= or SELECT TABLE= parameters to cause those parameters to act on VSAM component names (the names of their format-1 DSCBs) instead of their cluster names. Specifying this parameter can eliminate the need for SAMS:Disk to access the catalog to retrieve the cluster name for VSAM components that do not meet the SELECT DSNAMES= or TABLE= list. This saves significant processing time and also prevents many “broken” VSAM cluster messages (those that refer to components outside of the range of the DSNAMES= or TABLE= parameters).

This COMPONENTS parameter is only allowed on SELECT DSNAMES= or SELECT TABLE= commands.

This COMPONENTS parameter is ineffective under SCAN CATALOGS= or FIND commands and when checking fields that are not available within the FMT1 DSCB. This parameter should only be used with those SELECT CRITERIA fields that search the FMT1 information, NOT those parameters that review fields contained in the catalog or files data set, such as MODIFIED or ARCDT. The COMPONENT parm is intended for use mainly with the FMT1 DSCB date fields, (such as F1CREDT, F1MODDT) or if the SELECT CRITERIA= parameter is not used.

Use this parameter if you have installed DFP release 3.1 or above, have installed a VSAM allocation manager product or have taken administrative action to ensure that VSAM components have names that relate to their associated cluster names, or when not processing VSAM data sets.

For example, to report on format-1 DSCBs starting with VSAM created today on your VSAM volumes:

```
//jobname JOB (acct,info),etc.
//STEP EXEC DMS
SCAN REALVOLS
  SELECT DSNAMES=VSAM/ ,VOL=VSAM/ ,
        COMPONENTS,CRITERIA=( F1CREDT,EQ,TODAY )
  REPORT MVD,COMPONENTS
```

Figure 2-1. Sample DSCL REPORT Command

TABLE=

Specify the name of a member in the parmlib data set containing the list (table) of fully qualified data set names or patterns.

Note: Data sets are not automatically selected because of their appearance in this list. If any other parameters are specified on the same SELECT command, all parameters must be true for the data set to be selected. For example, in the following

command, the data set must both be listed in the member named “IMSDSNS” and must have been backed up within the last five days:

```
SELECT TABLE=IMSDSNS,CRITERIA=(ARCDT,GE,TODAY-5),DSN= /
```

Because a SELECT TABLE= statement is already very restrictive, you may not specify the VTOCS, nor CATALOGS= parameters on the same SELECT statement. You may, however, add the DSNAMES=, COMPONENTS, VOLUMES= or CRITERIA= parameters.

VOLUMES=

If you are running under the FIND command or you let SAMS:Disk sysparm VOL-SELRE default, you may optionally specify a list of volumes to be selected. If you are running under the SCAN command and have SAMS:Disk sysparm VOL-SELRE specified with a value of Y, you must specify this parameter.

You may specify up to 30 volumes or volume patterns.

The reason for this option is that the SELECT and EXCLUDE commands control the volumes that are scanned. If any SELECT command is specified without the VOLUMES= parameter, all volumes are scanned, regardless of any other SELECT commands. If multiple SELECT commands are entered and you want them to all process the same volumes, the volume list must be copied to each SELECT, as in the example below:

```
SCAN REALVOLS
SELECT VOL=(LABS/,WORK/),CRITERIA=(TRKS,GT,200),DSN= /
SELECT VOL=(LABS/,WORK/),CRITERIA=(EXTENTS,GT,3),DSN= /
REPORT SVDICT,MVDICT
```

You may add the DSNAMES=, COMPONENTS, TABLE=, VTOCS, CATALOGS= or CRITERIA= parameters on the same SELECT statement.

VTOCS

Normally, DSCL processes user data sets only, and bypasses VTOCs, VTOC indexes and catalogs. Specify this parameter to have SAMS:Disk process VTOCs and VTOC indexes. Because a SELECT VTOCS statement is already very restrictive, you may not specify the DSNAMES=, COMPONENTS, TABLE=, CATALOGS= or CRITERIA= parameters on the same SELECT statement. You may, however, add a VOLUMES= parameter.

Note: The command “SCAN REALVOLS” is required for a SELECT VOL=/,VTOCS to backup vtocs.

CATALOGS=

Normally, DSCL processes user data sets only, and bypasses VTOCs, VTOC indexes and catalogs. Specify this parameter to have SAMS:Disk process catalogs as specified by a fully qualified data set name or SAMS:Disk pattern. Because a SELECT CATALOGS= statement is already very restrictive, you may not specify the DSNAMES=, COMPONENTS, TABLE=, VTOCS nor CRITERIA= parameters

on the same SELECT statement. You may, however, add a VOLUMES= parameter.

CRITERIA=

Specify the selection criteria to be met before any SAMS:Disk action will occur. The criteria parameter is specified by a minimum of three subparameters as follows:

```
(operand1,operator,operand2)
```

where the operands may consist of an expression as defined below. The maximum number of subparameters allowed is controlled by sysparm VSCRIMAX. Because a SELECT CRITERIA= statement is already very restrictive, you may not specify the VTOCS nor CATALOGS= parameters on the same SELECT statement. You may, however, add the DSNAMES=, COMPONENTS, TABLE= or VOLUMES= parameters.

For logical operators, you may specify any of the following:

Table 2-3. Criteria Operators and their Meanings

Operator	Meaning
EQ	equal to
NE	not equal to
LT	less than
LE	less than or equal to
GT	greater than
GE	greater than or equal to

For operands, you may specify any of the following:

- Decimal number constants, such as 10 or 1000.
- Date constants in any acceptable SAMS:Disk format, such as 96200, 96.200, 96/200, 19JUL96, 07/19/96, 1996200, 1996.200, 19JUL1996, 1996/200 or 07/19/1996. If SAMS:Disk has trouble differentiating between a digit Julian date constant, such as 96200, and a decimal number constant, it will issue an appropriate message and ask you to use another format.
- Constant values TRUE, FALSE, YES, NO, ON or OFF. TRUE, YES, and ON are all equivalent. FALSE, NO, and OFF are all equivalent.
- Any of the DSCL select criteria field names listed below.

- Expressions that use addition (+) or subtraction (-) to combine a constant value with a field name or other constant value. For example, the expression TODAY-30 represents the date 30 days ago.

The following list shows the DSCL select criteria field name, listed alphabetically, with all date fields presented first. Then after a general discussion on dates, each criteria is described more fully.

For your convenience, the page number where the Field Name description can be found is also listed.

Table 2-4. DSCL Selection Criteria Field Names

Field Name	Page #	Definition
ARCDT	50	Last backup/archive date
CREDIT	50	Creation date
EXPDT	51	Expiration date
MODDT	51	Last modification date
TODAY	52	Today's date
USEDT	52	Last use date
F1CREDIT	53	Format 1 DSCB creation date
F1EXPDT	55	Format 1 DSCB expiration date
F1MODDT	54	Format 1 DSCB last mod
F1USEDT	54	Format 1 DSCB last use date
ALLOBLKS	55	Allocated in blocks
ALLOCYLS	55	Allocated in cylinders
ALLOPRI	56	Primary allocation amount
ALLOSEC	56	Secondary allocation amount
ALLOTBKS	56	Allocated in tracks
BLKSZ	56	Block size
CASPLITS	57	Control Area splits
CATSTAT	57	Catalog status of the data set
CISPLITS	58	Control Interval splits
DSORG	58	Data set organization

Field Name	Page #	Definition
EXTENTS	58	Extents in the data set
GDG	59	Generation data group generation
GDGGEN	59	Relative generation number
GDGSTAT	59	Generation data group status
IDLETRKS	59	Idle tracks in the data set
KBYTES	60	Kilobytes allocated to data set
LRECL	60	Logical record length
MODBIT	60	Modification (change) bit
MODELDCB	60	Model DSCB
MODIFIED	61	Modified since last backup
PASSWORD	61	Password-protected
PASSTYPE	62	Password protection type
PCTUSED	62	Percent used
PDSORG	63	Partitioned data set organization
PSORG	63	Physical Sequential data set organizations
RACFIND	64	RACF indicated
RECFM	64	Record format
SVCMODE	65	SVC mode field for VTOC statistics
TEMP	65	System temporary non-VSAM data set
TRKS	65	Tracks allocated
UNMOVE	66	Unmovable attribute
VOLCNT	66	data set allocated volume count
VOLSEQNO	66	Volume sequence number for multivolume data sets
VSAMORG	66	VSAM organization type

Midnight “Rapid Aging” Problems

When SAMS:Disk jobs that examine date fields are run very shortly after midnight, you must take into account the “rapid aging” that occurs at the stroke of midnight. Data sets that may have been created or used merely minutes ago suddenly appear as if the action took place a day ago!

When batch jobs create data sets, the create date used by the operating system is taken from the date the job started, not the date that each specific step of the job started. Hence data sets actually created in steps running after midnight, but part of a multistep job that began before midnight, will have create dates of the previous day — and immediately appear to be one day old!

Any selection tests based upon date values should take these two items into account, or many more data sets may be selected than were intended.

DSCL SELECT CRITERIA Field Name Descriptions

ARCDT — Last BACKUP/ARCHIVE date (date format)

Comparison dates may be entered in any acceptable SAMS:Disk format.

This field is set by inspecting the SAMS:Disk Files data set for the most recent backup or archive for the data set or cluster. If the SAMS:Disk Files data set does not have a record of a backup or archive, this field contains 00/00/0000, or 0.

For example, to select data sets never backed up by SAMS:Disk, use:

```
SELECT CRITERIA=( ARCDT, EQ, 0 )
```

To select data sets and clusters not backed up in the last 28 days, use `SELECT CRITERIA=(ARCDT,LE,TODAY-28)`. Note that this will select data sets and clusters never backed up as well!

Note: Because of the need to inspect the SAMS:Disk Files data set, using this parameter can cause SAMS:Disk processing time to increase substantially.

CREDIT — Creation date (date format)

Comparison dates may be entered in any acceptable SAMS:Disk format. For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of CREDIT is set to the value of the creation date field DS1CREDIT in the Format-1 DSCB for each data set or component. Missing or invalid values are assigned a value of 00/00/0000 or 0.

For all other processing of VSAM clusters, the value of CREDIT is set to the creation date field DSETCRDT in the catalog entry for each cluster. Missing or invalid values are assigned a value of 00/00/0000 or 0.

For example, to select data sets without creation dates, use:

```
SELECT CRITERIA=(CREDIT,EQ,0)
```

To select data sets and clusters with creation dates in the last 3 days, use `SELECT CRITERIA=(CREDIT,GE,TODAY-3)`. Note that this will NOT select data sets and clusters without creation dates!

For improved performance, review the criteria description for *FICREDIT* on page 53.

For related information, turn to the topic "*Midnight Rapid Aging Problems*" on page 49.

EXPDT — Expiration date (date format)

Comparison dates may be entered in any acceptable SAMS:Disk format.

For all processing of non-VSAM data sets, and reporting on VSAM components with "SCAN REALVOLS, REPORT COMPONENTS", the value of EXPDT is set to the value of the expiration date field DS1EXPDT in the Format-1 DSCB for each data set or component. Missing or invalid values are assigned a value of 00/00/0000 or 0.

For all other processing of VSAM clusters, the value of EXPDT is set to the expiration date field DSETEXDT in the catalog entry for each cluster. Missing or invalid values are assigned a value of 00/00/0000 or 0.

For example, to select data sets with expiration dates, use:

```
SELECT CRITERIA=(EXPDT,NE,0)
```

To select data sets and clusters set to expire in the next week, use:

```
SELECT CRITERIA=(EXPDT,NE,0,AND,EXPDT,LT,TODAY+7)
```

Note: The first part of the expression causes only data sets and clusters with valid expiration dates to be selected!

For improved performance, review the criteria description for *FIEXPDT* on page 55.

MODDT — Last modification date (date format)

Comparison dates may be entered in any acceptable SAMS:Disk format. For all processing of non-VSAM data sets, and reporting on VSAM components with "SCAN REALVOLS, REPORT COMPONENTS", the value of MODDT is set to the location pointed to by SAMS:Disk sysparm DSCBLMOD in the Format-1 DSCB for each data set or component. For more details, review the sysparm description for *DSCBLMOD* on page 136 of the *Systems Guide*. Missing or invalid values are assigned a value of 00/00/0000 or 0.

For all other processing of VSAM clusters, the value of MODDT is set to the last modified date in field AMDSTSP in the catalog entry for each cluster. Missing or invalid values are assigned a value of 00/00/0000 or 0.

For example, to select data sets without last modification dates, use:

```
SELECT CRITERIA=(MODDT,EQ,0)
```

To select data sets and clusters modified in the last day, use SELECT CRITERIA=(MODDT,GE,TODAY). Note that this will NOT select data sets and clusters without last modification dates!

XDM/RD and certain other applications bypass normal open processing and therefore do not maintain a last modified date. Thus, these data sets will never be selected by the above selection criteria.

For improved performance, review the criteria description for *FIMODDT* on page 54.

For related information, turn to the topic "*Midnight Rapid Aging Problems*" on page 49.

TODAY — Today's date (date format)

Comparison dates may be entered in any acceptable SAMS:Disk format.

This value is derived from the system clock.

USEDT — Last Use date (date format)

Comparison dates may be entered in any acceptable SAMS:Disk format. For all processing of non-VSAM data sets, and reporting on VSAM components with "SCAN REALVOLS, REPORT COMPONENTS", the value of USEDT is set to the location pointed to by SAMS:Disk sysparm DSCBLUSD in the Format-1 DSCB for each data set or component. For more details, see the sysparm description for *DSCBLUSD* on page 136 of the *Systems Guide*.

For all other processing of VSAM clusters, the value of USEDT is set to the location pointed to by SAMS:Disk sysparm DSCBLUSD in the Format-1 DSCB for the first data component of each cluster. But this field is filled in only for clusters cataloged in DMF/EF Type-2 catalogs. Consult the appropriate personnel at your own installation to determine if last use dates are being maintained by either Hitachi routines or the enhanced routines supplied with SAMS:Disk. If they are not, you should not use this field name for VSAM clusters.

To ensure that data sets without a last use date do not get archived when last use criteria are specified, invalid values are assigned a value of 2155366, a high date. To maintain compatibility, the old invalid date of 31DEC1999 or 99365 will be accepted until the cutoff date of 01JAN1999 or 99.001. The 99365 date will be converted to 2155366 if it is the object of USEDT or

F1USEDT comparisons. The NEW invalid USEDTE or F1USEDTE value is 0, 'zero', or 2155366 or 2155.366. Either of these values may be used now to search for zero and invalid USEDTEs.

Note: We recommend conversion begin immediately, so that you may begin treating USEDTEs of 99365 as 'real' beginning on 99.001. This will also assist you during testing with IPL dates GE 99.001.

For example, to select data sets with zero and invalid last use dates, use:

```
SEL CRI=(USEDTE,EQ,31DEC1999) - works until IPL dates GE 99.001
SEL CRI=(USEDTE,EQ,0)          - works now and in future
SEL CRI=(USEDTE,EQ,2155.366)   - works now and in future
```

To select data sets and clusters cataloged in ICF catalogs with date stamping turned on but not used in the last 30 days, use:

```
SELECT CRITERIA=(USEDTE,LE,TODAY-30)
```

Note: This will NOT select data sets and clusters without last use dates!

For improved performance, review the criteria description for *F1USEDTE* on page 54.

For related information, turn to the topic "*Midnight Rapid Aging Problems*" on page 49.

F1CREDT — Format-1 DSCB create date (date format)

Comparison dates may be entered in any acceptable SAMS:Disk format. For all processing of all data sets, the value of F1CREDT is set to the value of the creation date field DS1CREDT in the format-1 DSCB of each data set or VSAM component. Missing or invalid values are assigned a value of 00/00/0000 or 0.

Under "SCAN REALVOLS", this F1CREDT selection criteria will give increased performance over the CREDIT selection criteria due to eliminating the need for SAMS:Disk to access the catalog to retrieve the cluster name for VSAM components that do not meet this CREDIT test.

The creation date in the format-1 DSCBs of components of VSAM clusters may provide different results compared to selection based on the creation date maintained in the cluster's catalog entry. Non-VSAM data sets are processed identically between CREDIT and F1CREDT.

If you want to select VSAM clusters by the creation date in the cluster's catalog entry, while maintaining non-VSAM compatibility, use CREDIT instead of F1CREDT.

This selection criteria is only valid when "SCAN REALVOLS" is used. It is ignored for "SCAN CAT", unlike the CREDIT selection criteria, which is valid for both. Another difference between F1CREDT and CREDIT se-

lection criteria is that F1CREDT may allow your SAMS:Disk job to process more quickly. This faster processing occurs because F1CREDT is set up to allow SAMS:Disk to bypass some catalog processing.

For related information, turn to the topic "*Midnight Rapid Aging Problems*" on page 49.

F1MODDT — Format-1 DSCB last modified date (date format)

Comparison dates may be entered in any acceptable SAMS:Disk format.

For all processing of all data sets, the value of F1MODDT is set to the location pointed to by SAMS:Disk sysparm DSCBLMOD in the format-1 DSCB of each data set or VSAM component. For more details, review the sysparm description for *DSCBLMOD* on page 136 of *Systems Guide*. Missing or invalid values are assigned a value of 00/00/0000 or 0. The last modified date is not set in the format-1 DSCB of VSAM index components by SAMS:Disk or Hitachi.

Under "SCAN REALVOLS", this F1MODDT selection criteria will give increased performance over the MODDT selection criteria due to eliminating the need for SAMS:Disk to access the catalog to retrieve the cluster name for VSAM components that do not meet this MODDT test.

The modification date in the format-1 DSCBs of components of VSAM clusters may provide different results compared to selection based on the modification date maintained in the cluster's catalog entry. Non-VSAM data sets are processed identically between MODDT and F1MODDT.

If you want to select VSAM clusters by the modification date in the cluster's catalog entry, while maintaining non-VSAM compatibility, use MODDT instead of F1MODDT. This selection criteria is only valid when using "SCAN REALVOLS", it is ignored for "SCAN CAT", unlike the MODDT selection criteria which is valid for both. Another difference between F1MODDT and MODDT selection criteria is that F1MODDT may allow your SAMS:Disk job to process more quickly. This is because F1MODDT is set up to allow SAMS:Disk to bypass some catalog processing.

For related information, turn to the topic "*Midnight Rapid Aging Problems*" on page 49.

F1USEDT — Format-1 DSCB last used date (date format)

Comparison dates may be entered in any acceptable SAMS:Disk format. For all processing of all data sets, the value of F1USEDT is set to the location pointed to by SAMS:Disk sysparm DSCBLUSD in the format-1 DSCB of each data set or VSAM component. For more details, review the sysparm description for *DSCBLUSD* on page 136 in the *Systems Guide*. Missing or invalid values are assigned a value of 00/00/0000 or 0. The last used date is not set in the format-1 DSCB of VSAM index components.

Under “SCAN REALVOLS”, this F1USEDT selection criteria will give increased performance over the USED T selection criteria due to eliminating the need for SAMS:Disk to access the catalog to retrieve the cluster name for VSAM components that do not meet this USED T test.

This selection criteria is only valid when using “SCAN REALVOLS”, it is ignored for “SCAN CAT”, unlike the USED T selection criteria which is valid for both. Another difference between F1USED T and USED T selection criteria is that F1USED T may allow your SAMS:Disk job to process more quickly. This is because F1USED T is set up to allow SAMS:Disk to bypass some catalog processing.

For related information, turn to the topic *“Midnight Rapid Aging Problems”* on page 49.

F1EXPDT — Format-1 DSCB expiration date (date format)

Comparison dates may be entered in any acceptable SAMS:Disk format. For all processing of non-VSAM data sets and VSAM components with “SCAN REALVOLS”, the value of F1EXPDT is set to the value of the expiration date field DS1EXPDT in the format-1 DSCB for each data set or VSAM component. Missing or invalid values are assigned a value of 00/00/0000 or 0.

This selection criteria is only valid when using “SCAN REALVOLS”, it is ignored for “SCAN CAT”, unlike the EXPDT selection criteria which is valid for both. Another difference between F1EXPDT and EXPDT selection criteria is that F1EXPDT may allow your SAMS:Disk job to process more quickly. This is because F1EXPDT is set up to allow SAMS:Disk to bypass some catalog processing.

For related information, turn to the topic *“Midnight Rapid Aging Problems”* on page 49.

ALLOBLKS — Allocated in blocks (yes/no)

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of ALLOBLKS is set to YES/TRUE/ON if the DS1AVR bit is set and the DS1TRK bit is not set in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of ALLOBLKS is set to YES/TRUE/ON if the DS1AVR bit is set and the DS1TRK bit is not set in the space options field SPACOPTN in the catalog entry for each cluster.

ALLOCYLS — Allocated in cylinders (yes/no)

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value

of ALLOCYLS is set to YES/TRUE/ON if the DS1CYL bits are set in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of ALLOCYLS is set to YES/TRUE/ON if the DS1CYL bits are set in the space options field SPACOPTN in the catalog entry for each cluster.

ALLOPRI — Primary allocation amount (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of ALLOPRI is set to the primary allocation amount (like the “2” in “SPACE=(TRKS,(2,1))” as determined by the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of ALLOPRI is set to the primary space allocation (like the “2” in “TRACKS (2,1)”) field PRIMSPAC in the catalog entry for each cluster.

ALLOSEC — Secondary allocation amount (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of ALLOSEC is set to the secondary allocation amount (like the “1” in “SPACE=(TRKS,(2,1))” as determined by the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of ALLOSEC is set to the secondary space allocation (like the “1” in “TRACKS (2,1)”) field SCONSPAC in the catalog entry for each cluster.

ALLOTRKS — Allocated in tracks (yes/no)

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of ALLOTRKS is set to YES/TRUE/ON if the DS1TRK bit is set and the DS1CYL bits are not set in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of ALLOTRKS is set to YES/TRUE/ON if the DS1TRK bit is set and the DS1CYL bits are not set in the space options field SPACOPTN in the catalog entry for each cluster.

BLKSZ — Block size (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of BLKSZ is set to the value of the field DS1BLKL in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of BLKSZ is set to the physical blocksize field PHYBLKSZ or the CI size field AMDCINV in the catalog record for each VSAM cluster, depending on the value of sysparm VSUSEBLK. For more information, review the sysparm description for VSUSEBLK on page 192 in the *Systems Guide*.

CASPLITS — Control Area splits (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of CASPLITS is set to 0. For all other processing of VSAM clusters, the value of CASPLITS is set to the Number of Control Area Splits field AMDNCAS in the catalog entry for each cluster.

CATSTAT — Catalog status of the data set

This field should be used only when running SCAN REALVOLS. The second operand in a CATSTAT comparison may be any of the following:

Table 2-5. CATSTAT Values and their Meanings

Value	Meaning
YES/TRUE/ON	Data set is properly cataloged
NO/FALSE/OFF	Data set is not cataloged (except CATOTHER or WRONGVOL)
WRONGVOL	Data set is cataloged, but to the wrong volume
CATOTHER	Catalog status could not be determined for the data set due to an error, such as the volume with the catalog is not online

For example, to select non-cataloged and mis-cataloged data sets, use:

```
SELECT CRITERIA=( CATSTAT,NE,YES )
```

Data sets with a CATOTHER status will never be selected unless (CATSTAT,EQ,CATOTHER) is specified. This insures that if a catalog is damaged or offline, data sets won't be selected by any other combinations of the CATSTAT values.

Note: If you are not running a report that requires the catalog status to be checked, specifying this criteria can cause a SIGNIFICANT increase in both elapsed and CPU times (since the catalog is checked for every data set).

Caution: Since some system data sets by definition remain uncataloged, use caution when selecting data sets for processing based on their catalog status. For example, a common practice is to catalog all data sets residing on an alternate sysres to volser (*****). If the DSCL criteria statement

(CATSTAT,EQ,WRONGVOL) all data sets on the alternate sysres will be selected for processing.

CISPLITS — Control Interval splits (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of CISPLITS is set to 0. For all other processing of VSAM clusters, the value of CISPLITS is set to the Number of Control Interval Splits field AMDNCIS in the catalog entry for each cluster.

DSORG — Data set organization

The second operand in a DSORG comparison may be any of the following:

Table 2-6. DSORG Criteria Field Values

Value	Definition
PS	Sequential
SEQ	Sequential
PO	Partitioned
PDS	Partitioned
DA	Direct access
VSAM	VSAM
AM	Access Method (VSAM)
UNKNOWN	no DSORG bits on in format-1 DSCB
INVALID	invalid pattern of DSORG bits in format-1 DSCB

For example, to select all partitioned and sequential data sets, use:

```
SELECT CRITERIA=(DSORG,EQ,PO,OR,DSORG,EQ,PS)
```

To select unmovable data sets by using the UNMOVE field. For example,

```
SELECT CRITERIA=(DSORG,EQ,PS,AND,UNMOVE,EQ,YES)
```

For similar fields, review the criteria descriptions for *VSAMORG* (on page 66), *PDSORG* (on page 63), and *PSORG* (on page 63).

EXTENTS — extents in the data set (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of EXTENTS is set to the value of the DS1NOEPV field in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of EXTENTS is set to the Number of Extents field NOEXTNT in the catalog entry for the first data component of each cluster. It does not include the extents in the index component of any cluster. VSAM cluster reports, however, will display the sum of the two, which thus will be larger than the number on which selection was based.

For a similar field, review the criteria description for MODEL DSC on page 60.

GDG — Generation data group generation (yes/no)

For non-VSAM data sets, the value of this field is set to YES/TRUE/ON if the name represents, and is cataloged as, a generation data group generation.

For VSAM data sets, the value of GDG is always set to NO/FALSE/OFF.

GDGGEN — Relative generation number (number)

For non-GDG data sets (see the above field GDG), the value of this field is set to 0. For GDG data sets, this field is set to the relative generation number as shown by the catalog.

For example, to select GDG generations older than the three most current versions (older than the -2, -1 and 0 generations) use:

```
SELECT CRITERIA=(GDGGEN,LE,-3)
```

Because non-GDGs are given a relative generation number of 0, to select the most current version of GDGs, use:

```
SELECT CRITERIA=(GDG,EQ,YES,AND,GDGGEN,EQ,0)
```

GDGSTAT — Generation data group status

The value of this field represents the status of a generation data group. The second operand in a GDGSTAT comparison may be any of the following:

Table 2-7. GDGSTAT Criteria Field Values

Value	Definition
YES/TRUE/ON	active GDG
NO/FALSE/OFF	not a GDG
ROLLOFF	rolled off GDG
DEFERRED	deferred GDG

For VSAM data sets, the value of GDGSTAT is always set to NO/FALSE/OFF.

IDLETRKS — Idle tracks (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of IDLETRKS is set to the total number of tracks occupied by the data set or component as shown by its Format-1 and possible Format-3 DSCBs, less the value of the DS1LSTAR field in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of IDLETRKS is set to the number of tracks represented by the difference between the High Allocated RBA field HARBAS and the High Used RBA HURBAS in each cluster’s catalog entry.

KBYTES — Kilobytes allocated to the data set (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of KBYTES is set to the total number of tracks occupied by each data set or component as shown by its format-1 and possible format-3 DSCBs, multiplied by the maximum blocksize for the device, and divided by 1000.

For all other processing of VSAM clusters, the value of KBYTES is set to the number of tracks represented by the each of the extents listed in each cluster’s catalog entry, multiplied by the maximum blocksize for the device, and divided by 1000.

LRECL — Logical record length (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of LRECL is set to the value of the logical record length field DS1LRECL in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of LRECL is set to the average logical record length field LRECL in the catalog entry for each cluster.

MODBIT — Modification (change) bit (on/off)

For all processing of non-VSAM data sets, DMF/EF Type-2 VSAM clusters, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of MODBIT is set to YES/TRUE/ON if the change bit DS1IND02 is set in the Format-1 DSCB for each data set or component. Same as the MODIFIED field.

Note: Certain applications bypass normal OPEN processing. Data sets updated by these applications will not have their change bits set. Thus, these data sets will never be selected by SEL CRI=(MODBIT,EQ,YES). To select all potentially modified data sets, you must specifically select these data sets.

MODELDC — Is the data set a model DSCB? (yes/no)

For non-VSAM data sets, the value of MODELDC is set to YES/TRUE/ON if the value of the DS1NOEPV field is 0 and the value of the DS1SCAL3 field is 0 in the Format-1 DSCB for the data set.

For VSAM data sets, the value of MODELDC is always set to NO/FALSE/OFF.

For a similar field, review the criteria description for EXTENTS on page 58.

MODIFIED — Modified since last backup (yes/no)

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of MODIFIED is set to YES/TRUE/ON if the change bit DS1IND02 is set in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of MODIFIED is set to YES/TRUE/ON if the SAMS:Disk Files data set shows that this cluster has never been backed up or archived by SAMS:Disk, or if the last modified date and time in field AMDSTSP in the catalog entry for each cluster are more recent than the last SAMS:Disk backup or archive of the clusters, or if the VSAM cluster open flag in catalog field OPENIND is on. The OPENIND flag is set ON whenever a VSAM cluster is open and set OFF when it is closed. If the cluster is not closed normally (i.e., when a system crash occurs), then its flag is left on. Even if the comparison of last modified date/time and last backup date/time indicates the data set has not been modified since its last backup, the OPENIND flag test will indicate that MODIFIED is YES/TRUE/ON.

Note: If a VSAM cluster is currently in use, normal system enqueues will ensure that SAMS:Disk does not get access to the data set when the OPENIND is ON.

Because SAMS:Disk must check the files data set for each cluster, using this parameter with VSAM clusters may increase processing time.

Note: Certain applications bypass normal OPEN processing. Non-VSAM data sets updated by these applications will not have their change bits on. These data sets will never be selected by the following SELECT statement:

```
SELECT CRITERIA=(MODIFIED,EQ,YES)
```

To select all potentially modified data sets, you must specifically select these data sets.

VSAM clusters used by these applications (i.e., XDM/RD) will have a last modified date of 00/00/0000 or 0. In this case, the last use date for the primary data component will be used in comparison with the date of the last backup. The change bit for the primary data component must also be on.

PASSWORD — Password-protected (yes/no)

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of PASSWORD is set to YES/TRUE/ON if the DS1IND10 bit is set in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of PASSWORD is set to YES/TRUE/ON if the password field PASSWALL in each cluster’s catalog entry shows that a read, write, control or master password exists.

PASSTYPE — Password protection type

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of PASSWORD is set to READ or WRITE depending on the values of the DS1IND10 and DS1IND04 bits in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of PASSWORD is set to READ, WRITE, CINV or MSTR depending on the value of the password field PASSWALL in each cluster’s catalog entry.

The second operand in a PASSTYPE comparison may be:

Table 2-8. PASSTYPE Criteria Field Values

Value	Definition
READ	Read password required
WRITE	Write password required
CINV	Control interval password (VSAM only)
MSTR	Master level password required (VSAM only)

For example, to select all write-password-protected data sets, use:

```
SELECT CRITERIA=(PASSTYPE,EQ,WRITE)
```

“Greater than” and “less than” type comparisons may also be made, based upon read-level password authority being less than write, write less than control interval, and control interval less than master. For example, to select all data sets with write, control interval, or master level password protection, use:

```
SELECT CRITERIA=(PASSTYPE,GE,WRITE)
```

PCTUSED — Percent used (number)

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of PCTUSED is set to 0 if the DS1LSTAR field in the Format-1 DSCB for

each data set or component is set to 0. If the DS1LSTAR field is not 0, this PCTUSED is set to the number of tracks in the DS1LSTAR divided by the tracks allocated to the data set, expressed as a percentage with a minimum of 1 and a maximum of 100.

This means that a 1,000 track non-VSAM data set that uses 1 track would have a PCTUSED of 1, not 0. If it were truly empty (a zero DS1LSTAR), it would have a PCTUSED of 0. Beware of data sets used by applications that do not maintain the DS1LSTAR field.

Note: PO data sets that have a member count of zero will not be selected, even if you specify "PCTUSED,EQ,0", because the directory information makes the data set appear partially used.

VSAM components never appear to be empty.

For all other processing of VSAM clusters, the value of PCTUSED is set to 0 if the value of the High Used RBA field HURBADs in each cluster's catalog entry is 0. If the HURBA is not 0, this PCTUSED is set to the High Used RBA HURBADs divided by the High Allocated RBA HARBADs, expressed as a percentage with a minimum of 1 and a maximum of 100.

This means that a cluster that is nearly empty, or which has had records inserted and subsequently deleted, would have a PCTUSED of 1, not 0. If it were truly empty (a zero HURBADs), it would have a PCTUSED of 0.

PDSORG — Partitioned data set organization

Possible values for the second operand in a PDSORG comparison are:

Table 2-9. PDSORG Criteria Field Values

Value	Definition
PDS	Partitioned
PO	Partitioned
POE	Partitioned extended

For example, to select all PDSs, use:

```
SELECT CRITERIA=(PDSORG,EQ,PO) or
SELECT CRITERIA=(PDSORG,EQ,PDS)
```

PSORG — Physical Sequential Extended

Possible values for the second operand in a PSORG comparison are:

Table 2-10. PSORG Criteria Field Values

Value	Definition
PS	Sequential
SEQ	Sequential
SEQE	Sequential Extended (Striped data set)

For example, to select all PSs, use:

```
SELECT CRITERIA=(PSORG,EQ,PS) or
SELECT CRITERIA=(PSORG,EQ,SEQ)
```

RACFIND — RACF-indicated (yes/no)

For all processing of non-VSAM data sets, and reporting on VSAM components with “SCAN REALVOLS, REPORT COMPONENTS”, the value of RACFIND is set to YES/TRUE/ON if the DS1IND40 bit is set in the Format-1 DSCB for each data set or component.

For all other processing of VSAM clusters, the value of RACFIND is set to YES/TRUE/ON if the x'80' bit is set in the RACF security attributes field SECFLAGS in each cluster's catalog entry.

RECFM — Record format

The second operand in a RECFM comparison may be any of the following:

Table 2-11. RECFM Criteria Field Values

Value	Definition
F	Fixed-length records
FB	Fixed blocked records
FBA	Fixed blocked with ASA carriage control
FBM	Fixed blocked with machine carriage control
V	Variable-length records
VB	Variable blocked records
VBA	Variable blocked with ASA carriage control
VBM	Variable blocked with machine control
U	Undefined length records

This field requires an “exact match”; that is, `SEL CRI=(RECFM,EQ,FB)` will select only FB data sets, not F or FBA, etc.

If you need to select data sets with a particular RECFM attribute but you don’t care what other attributes it is combined with, use the following fields and simply test for the condition being present:

Table 2-12. RECFM Criteria Field Values using Patterns

Value	Definition
RECFMF	Fixed attribute (yes or no)
RECFMV	Variable attribute (yes or no)
RECFMU	Undefined attribute (yes or no)
RECFMS	Standard or Spanned attribute (yes or no)
RECFMT	Track overflow attribute (yes or no)
RECFMB	Blocked attribute (yes or no)
RECFMA	ASA carriage attribute (yes or no)
RECFMM	Machine carriage attribute (yes or no)

For example, `SELECT CRITERIA=(RECFMB,EQ,YES)` would select all blocked data sets, regardless if fixed or variable, etc.

`SELECT CRITERIA=(RECFMV,EQ,YES,AND,RECFMS,EQ,YES)` would select all variable-spanned data sets, regardless of whether they are blocked or not.

SVCMODE — SVC mode field for VTOC statistics (number)

The value of SVCMODE is set to the value of DS1SVCMO, at the offset defined by sysparm DSCBSVMD, within the format-1 DSCB for the data set. This field is maintained in the format-1 DSCB by the SAMS:Disk Open SVC as described on page 31 in the *Installation Guide*.

TEMP — System temporary non-VSAM data set (yes/no)

For non-VSAM data sets, the value of this field is set to YES/TRUE/ON if the data set name matches SAMS:Disk pattern name `SYS?????.T?????.?????./` or `***SYSUT` followed by anything.

For VSAM data sets, the value of TEMP is always set to NO/FALSE/OFF.

TRKS — Tracks allocated

For all processing of non-VSAM data sets, and reporting on VSAM components with `“SCAN REALVOLS, REPORT COMPONENTS”`, the value

of TRKS is set to the total number of tracks occupied by each data set or component as shown by its format-1 and possible format-3 DSCBs.

For all other processing of VSAM clusters, the value of TRKS is set to the number of tracks represented by each of the extents listed in each cluster's catalog entry.

UNMOVE — Unmovable attribute (yes/no)

For all processing of non-VSAM data sets, and reporting on VSAM components with "SCAN REALVOLS, REPORT COMPONENTS", the value of UNMOVE is set to YES/TRUE/ON if the unmovable attribute bit DS1DSGU is set in the Format-1 DSCB for each data set or component. For related information, review the criteria description for DSORG on page 58.

For all other processing of VSAM clusters, the value of UNMOVE is always set to NO/FALSE/OFF.

VOLCNT — Data set allocated volume count (number)

The value of VOLCNT is set to the number of allocated volumes for the data set. For non-VSAM cataloged data sets the volume count is retrieved from the catalog. The VOLCNT value is set to 1 for non-VSAM uncataloged data sets. The volume count for the VSAM cluster is retrieved from the catalog for each VSAM data set.

VOLSEQNO — Multivolume data set Volume Sequence Number

For all processing of non-VSAM data sets, and reporting on VSAM components with "SCAN REALVOLS, REPORT COMPONENTS", the value of VOLSEQNO is set to the value of the DS1VOLSQ field in the Format-1 DSCB for each data set or component.

For example, `SELECT CRITERIA=(VOLSEQNO,GT,1)` will select all non-first portions of multivolume data sets.

For all other processing of VSAM clusters cataloged in DMF/EF Type-2 catalogs, the value of VOLSEQNO is set to the DS1VOLSQ field in the Format-1 DSCB for the first data component of each cluster. For VSAM clusters cataloged in non-DMF/EF Type-2 catalogs, this field is set to 1.

VSAMORG — VSAM organization type

The second operand in a VSAMORG comparison may be any of the following:

Table 2-13. VSAMORG Criteria Field Values

Value	Definition
KSDS	Key-sequenced data set (cluster)
ESDS	Entry-sequenced data set (cluster)
RRDS	Relative record data set (cluster)
SVSAM	Linear data set (cluster)
BROKEN	Broken VSAM data sets (component)

For example, to select only KSDS clusters, use:

```
SELECT CRITERIA=(DSORG,EQ,VSAM,AND,VSAMORG,EQ,KSDS)
```

When SAMS:Disk determines that a VSAM data set is broken, the value of VSAMORG is set to BROKEN. Broken VSAM data sets will only be selected for the REPORT action command. Only the DSINDEX, DSUTIL, MAPPER, MVD and SVD reports will report on broken VSAM data sets. The broken VSAM data sets are reported at the component level as opposed to the cluster level.

When using the SCAN REALVOLS command, uncataloged or incorrectly cataloged VSAM components will be detected as broken if the COMPONENTS parameter is not specified. When using the SCAN CAT or FIND commands, cataloged VSAM components that do not physically reside on their cataloged volumes will be detected as broken by SAMS:Disk.

Specifying the SAMS:Disk sysparm DSCLMSG to Y causes SAMS:Disk to issue messages concerning the broken VSAM data sets.

For related information, review the criteria description for *DSORG* (on page 58), *PDSORG* (on page 63), *PSORG* (on page 63).

DSCL Select Criteria Boolean (AND/OR) Logic

The Boolean logic support of the CRITERIA parameter is invoked by specifying either an AND or an OR between conditional tests. Remember that a conditional test consists of an operand, an operator, and a second operand. You may specify a maximum of 64 conditional tests on a single CRITERIA parameter when each is separated by a compound condition (note that the maximum allowed may be affected by sysparm VSCRIMAX). Therefore, a valid Boolean test would be:

```
SELECT CRITERIA=( TRKS,GT,300,AND,
                  PCTUSED,LT,80,AND,
                  CREDT,GT,TODAY-60 )
```

As in the example above, if all of the compound conditions are AND, all of the conditional tests specified must be true for the statement to be true. Therefore, the data set would have to be larger than 300 tracks AND be less than 80 percent used AND have a creation date within the last 60 days to meet the criteria. Conversely, if all of the compound conditions are OR, if any of the conditional tests are true, the statement is true.

If both AND OR compound conditions are specified on a single command, all AND conditions are logically separated by each OR. In other words, if all AND conditions are true and an OR condition is encountered, the statement is considered true. To better describe the tests involved, let's look at a few examples.

1. Suppose you would like to select all data sets that are either larger than 300 tracks and have a percent used of less than 65, or that are less than or equal to 300 tracks but have a percent used less than 50. The required command would be:

```
SELECT CRITERIA=( TRKS,GT,300,AND,PCTUSED,LT,65,OR,
                  TRKS,LE,300,AND,PCTUSED,LT,50 )
```

2. Suppose you want to select all data sets that are larger than 50 tracks and meet one of the following criteria: either the data set is expired or the last modification date was not within the last 90 days. The command that would accomplish this is:

```
SELECT CRITERIA=( TRKS,GT,50,AND,EXPDT,LT,TODAY,AND,
                  EXPDT,NE,0,
                  OR,
                  TRKS,GT,50,AND,MODDT,LT,TODAY-90 )
```

Remember that in DSCL, neither indentation nor line breaks are important to the command parser, but they can be used to make the meaning clearer to users.

Note in this example that the "TRKS,GT,50" test was specified on both sides of the OR condition. This is necessary because the OR condition logically separates the

AND conditions. See what happens when the second “TRKS,GT,50” test is removed:

```
SELECT CRITERIA=( TRKS,GT,50,AND,EXPDT,LT,TODAY,AND,
                  EXPDT,NE,0,
                  OR,
                  MODDT,LT,TODAY-90)
```

With this test, data sets will be selected if:

- tracks are greater than 50 AND the expiration date is less than today’s date AND the expiration date is not zero, OR if
- the data set has not been modified in the last 90 days.

Note that the second test now makes no comparisons for track size and will be selected regardless of its size if the modification date test is true.

- To select all data sets spanning more than 400 tracks:

```
SELECT CRITERIA=( TRKS,GT,400)
```

- Select all data sets created since January 9, 1996:

```
SELECT CRITERIA=( CREDT,GT,1/9/1996)
```

- Select all data sets that have not been modified within the last year:

```
SELECT CRITERIA=( MODDT,LT,TODAY-365)
```

- Select all data sets that are greater than 300 tracks and that have not been modified in the last 60 days:

```
SELECT CRITERIA=( TRKS,GT,300,AND,MODDT,LT,TODAY-60)
```

- Select all data sets that either do not have a valid last use date, or that have not been used in the last year:

```
SELECT CRITERIA=( USEDTE,EQ,12/31/1999,OR,
                  USEDTE,LT,TODAY-365)
```

EXCLUDE Command

The EXCLUDE command allows you to exclude data sets from processing. In order for a data set to be excluded, all tests specified on a single EXCLUDE command must be met. If only one parameter is specified, only that single test must be met for the exclusion to be honored. If the EXCLUDE command is specified more than once for any given ACTION command, it has the effect of a logical “OR” between EXCLUDE commands. That is, if any EXCLUDE command is true, then the data set is bypassed, regardless if other EXCLUDE commands test true or not.

For example, if the DSNAMES= parameter or the VOLUMES= parameter, but not both, are specified, conceptually think “don’t process these data sets” or “don’t process these volumes”. If both the DSNAMES= and VOLUMES= parameters are specified, conceptually think “don’t process these data sets if they reside on these volumes”.

Note: If a SELECT command immediately precedes or follows any EXCLUDE command, special rules apply. Refer to the heading entitled “*SELECT and EXCLUDE Rules*” on page 31 for a further explanation.

The syntax of the EXCLUDE command, is as follows:

```
EXCLUDE      DSNAMES= , COMPONENTS , TABLE= , VOLUMES= ,
             VTOCS , CATALOGS= , CRITERIA=
```

DSNAMES=

You may specify the data sets to be excluded by NAME or PATTERN. Up to ten names or patterns are allowed.

Because an EXCLUDE DSNAMES= statement is already very restrictive, you may not specify the VTOCS or CATALOGS= parameters on the same EXCLUDE statement. You may, however, add the COMPONENTS, TABLE=, VOLUMES= or CRITERIA= parameters.

For improved performance for VSAM, see the COMPONENTS parameter description on page 44.

COMPONENTS

Under “SCAN REALVOLS”, the COMPONENTS parameter will modify the action of EXCLUDE DSNAMES= or EXCLUDE TABLE= parameters to cause those parameters to act on VSAM component names (the names of their format-1 DSCBs) instead of their cluster names. Specifying this parameter can eliminate the need for SAMS:Disk to access the catalog to retrieve the cluster name for VSAM components that do not meet the EXCLUDE DSNAMES= or TABLE= list. This saves significant processing time and also prevents many “broken” VSAM cluster messages (those that refer to components outside of the range of the DSNAMES= or TABLE= parameters).

This COMPONENTS parameter is only allowed on EXCLUDE DSNAMES= or EXCLUDE TABLE= commands.

This COMPONENTS parameter is ineffective under SCAN CATALOGS= or FIND commands.

Use this parameter if you have installed DFP release 3.1 or above, have installed a VSAM allocation manager product or have taken administrative action to ensure that VSAM components have names that relate to their associated cluster names, or are not processing VSAM data sets.

TABLE=

Specify the name of a member in the parmlib data set that contains a list of data set names and patterns that are to be excluded from SAMS:Disk processing.

Note: Data sets are not automatically excluded because of their appearance in this list. If any other parameters are specified on the same EXCLUDE command, all tests must be true for the data set to be excluded. For example, in the following command, the data set must both be listed in the member named “IMSDSNS” and be more than 300 tracks in size to be excluded:

```
EXCLUDE TABLE=IMSDSNS,CRITERIA=(TRKS,GT,300)
```

Because an EXCLUDE TABLE= statement is already very restrictive, you may not specify the VTOCS, nor CATALOGS= parameters on the same EXCLUDE statement. You may, however, add the DSNAMES=, COMPONENTS, VOLUMES= or CRITERIA= parameters.

VOLUMES=

You may exclude all data sets on a volume or volume group by use of this parameter. Up to 30 volumes or volume patterns are allowed.

When an EXCLUDE command with the VOLUMES= parameter is specified under a SCAN REALVOLS command, special processing rules apply. If the VOLUMES= parameter is specified by itself with no other parameters (such as CRITERIA= or DSNAMES=), the volume will not have its VTOC read, even if other SELECT commands are present that name the same volume or pattern name. Conversely, if any other parameter is specified on the EXCLUDE command, this will have no effect on which volumes get scanned to satisfy the request.

You may add the DSNAMES=, COMPONENTS, TABLE=, VTOCS, CATALOGS= or CRITERIA= parameters on the same EXCLUDE statement.

VTOCS

Normally, DSCL processes user data sets only, and bypasses VTOCs, VTOC indexes and catalogs. This parameter is optional, you may specify it to have SAMS:Disk bypass VTOCs and VTOC indexes.

Because an EXCLUDE VTOCS statement is already very restrictive, you may not specify the DSNAMES=, COMPONENTS, TABLE=, CATALOGS= nor CRITERIA= parameters on the same EXCLUDE statement. You may, however, add a VOLUMES= parameter.

CATALOGS=

Normally, DSCL processes user data sets only, and bypasses VTOCs, VTOC indexes and catalogs. This parameter is optional, you may specify it to have SAMS:Disk bypass catalogs as specified by a required fully qualified data set name or SAMS:Disk pattern.

Because an EXCLUDE CATALOGS= statement is already very restrictive, you may not specify the DSNAMES=, COMPONENTS, TABLE=, VTOCS nor CRITERIA= parameters on the same EXCLUDE statement. You may, however, add a VOLUMES= parameter.

CRITERIA=

Specify the exclusion criteria that, if met, will cause processing by SAMS:Disk for that data set to be bypassed. Refer to the parameter description for CRITERIA= on page 47 for details.

Because an EXCLUDE CRITERIA= statement is already very restrictive, you may not specify the VTOCS nor CATALOGS= parameters on the same EXCLUDE statement. You may, however, add the DSNAMES=, COMPONENTS, TABLE=, VOLUMES= parameters.

ACTION Commands

The commands REPORT, VREPORT, BACKUP, ARCHIVE, DELETE, VBACKUP, RELEASE, MOVE, COPY, EXPIRE, and BILLING are known as action commands because each causes SAMS:Disk to take a specific action against a selected data set or volume. You must provide at least one of these commands for each SCAN/FIND group.

- The REPORT and VREPORT commands and their parameters are described beginning on page 99.
- The BACKUP, ARCHIVE, DELETE, and EXPIRE commands and their parameters are described beginning on page 165.
- The VBACKUP command and its parameters are described beginning on page 251.
- The RELEASE command and its parameters are described beginning on page 353.
- The MOVE and COPY commands and their parameters are described beginning on page 331.
- The BILLING command is described beginning on page 385.

Complex DSCL Command Sequences

Even complicated structures are based on the simple rules discussed previously. In time, you will be able to build very sophisticated processing, in one or two scans if you like. An example:

```
//jobname JOB (acct,info),etc.
//*****
/* GET A BEFORE IMAGE REPORT, DELETE TRASH, ARCHIVE UNUSED  *
/* DATA SETS, BACK UP EVERYTHING USEFUL, RELEASE SPACE ON  *
/* OVERALLOCATED DATA SETS. THEN GET AN AFTER IMAGE REPORT *
//*****
//STEP EXEC DMS
SET MODE=SIMULATE
PASSWORD MASTER=the-DMS-master-password
SCAN REALVOLS
*****
* GET A BEFORE IMAGE REPORT: *
VREPORT ALLOCS,COMPONENTS
*****
* DELETE TRASH: *
*****
SELECT CRITERIA=(TEMP,EQ,YES,AND,CREDT,LE,TODAY-2)
SELECT CRITERIA=(CATSTAT,NE,YES,AND,CREDT,LE,TODAY-2)
DELETE
*****
* ARCHIVE MOST GDGS, LEAVING 2 GENERATIONS ON DISK: *
*****
EXCLUDE DSNAMES=(data.set.name.patterns.that.should,
have.more.than.two.generations,
left.on.disk),
SELECT CRITERIA=(GDG,EQ,YES,AND,GDGEN,LE,-2)
ARCHIVE DISP=RECAT,EXPDT=99000 catalog control
*****
* ARCHIVE OTHER GDGS, LEAVING 5 GENERATIONS ON DISK: *
*****
SELECT CRITERIA=(GDG,EQ,YES,AND,GDGEN,LE,-5)
ARCHIVE DISP=RECAT,EXPDT=99000 catalog control
*****
* ARCHIVE UNUSED STUFF FOR 7 YEARS, ALLOW AUTO-RESTORE: *
*****
SELECT CRI=(USEDT,LE,TODAY-28) not used in 28 days
EXCLUDE DSNAMES=SYS/ leave my SYS/ on disk
EXCLUDE DSNAMES=*.DSNDBC./ XDM/RD doesn't maintain,USEDT
ARCHIVE DISP=RECAT,RETPD=2557
*****
* DATA SET-ORIENTED BACKUP EVERYTHING THAT NEEDS IT: *
*****
SELECT CRITERIA=(MODIFIED,EQ,YES) newly modified
SELECT CRITERIA=(ARCDT,LE,TODAY-27) about to roll off
SELECT DSNAMES=*.DSNDBC./ XDM/RD doesn't maintain MODDT
SELECT CATALOGS=/ my DMF/EF Type-2 catalogs
SELECT VTOCS for RECOVER VOL=,VTOCS
BACKUP RETPD=28
*****
* RELEASE OVER-ALLOCATED SPACE: *
*****
SELECT CRITERIA=(DSORG,EQ,VSAM,AND,PCTUSED,LT,70)
RELEASE PCTUSED=75
```

Figure 2-2. Sample DSCL Command Structure

```
SELECT CRITERIA=(DSORG,EQ,PO,AND,PCTUSED,LT,75)
RELEASE PCTUSED=80
SELECT CRITERIA=(DSORG,EQ,PS,AND,PCTUSED,LT,100)
RELEASE PCTUSED=100
//*****
/* GET AN AFTER IMAGE REPORT:                      *
//*****
// EXEC DMS
    SCAN REALVOLS
    VREPORT ALLOCS
```


Chapter 3. Basic System — FMS PROC

All of the commands below are executed by the FMS PROC:

- EXCLUDE
- PASSWORD
- RECOVER
- SCAN
- SELECT
- SET

FMS processes both non-VSAM data sets and VSAM clusters. In its simplest form, the JCL required to use FMS would look like:

```
//jobname JOB (acct,info),..
//STEP    EXEC FMS
          FMS commands
```

Overview

The objective of adding the FMS PROC to SAMS:Disk was to provide a DSCL-like command language for data set recovery. This PROC provides equivalent functionality available in the original RECOVER PROC, and is equipped with its own palette of commands and parameters.

You also have a rich set of criteria to select data sets for and exclude data sets from processing as desired. SAMS:Disk preprocesses your command stream to find any overlapping requests, that is, those that request processing of the same volumes or catalogs.

To avoid unexpected execution of incomplete or partially bad command strings, the FMS PROC takes no action until all commands have been successfully understood.

To allow for sorted reports, the FMS PROC has separate process, sort and print steps. Sorted reports are also available in simulate mode.

Condition Codes

The FMS PROC returns the following JCL condition codes:

Table 3-1. FMS Condition Codes

Code	Meaning
0	Successful completion.
4	No data sets selected or 1 or more data sets not restored.
8	Data set(s) resource could not be acquired (for example, security or enqueues).
12	Unrecoverable error encountered in data set(s).
16	Command error.

Commands

The FMS PROC is controlled by commands provided to the SYSIN DD statement. The commands are of four main types:

1. **Environment commands.** Those commands that set up the environment in which SAMS:Disk will run: SET and PASSWORD.
2. **Candidate commands.** Those commands that find candidate data sets: SCAN.

3. **Selection commands.** Those commands that select or exclude data sets to be processed, based on your own specific criteria: SELECT and EXCLUDE.
4. **Action commands.** Those commands that take action upon the selected resources: RECOVER.

The minimum information required for you to execute FMS is a SCAN command to name the DSNINDEX records to be processed, a SELECT command, and an action command.

You may add other commands to gain additional selection or functionality.

FMS Command Sequence

FMS commands are listed below in their proper sequence. Each command is optional, except as described below.

Table 3-2. FMS Command Sequence

Type	Name	Parameters
Environment	SET	MODE=, GEN=, VTOCMSG=, MSGLEVEL=
Environment	PASSWORD (VSAM only)	CATALOG=, MASTER=
Candidate	SCAN	DSNINDEX
Selection	SELECT	DSNAMES=, TABLE=, VOLUMES=, VTOCS, CATALOGS=, CRITERIA=, TDSN=
Selection	EXCLUDE	DSNAMES=, TABLE=, VOLUMES=, VTOCS, CATALOGS=, CRITERIA=, TDSN=
Action	RECOVER	TOVOLUME=, POOL=, DISP=, NEWHLQ=, SCRATCH, CREATE, ERASE, NOLOAD, AIX=, TOCAT=, EXPORTF=, DEFALIAS=

General Rules Pertaining to the FMS Command Structure

- SET and PASSWORD commands are optional. If you do specify a PASSWORD command, it applies to processing of VSAM password-protected clusters only. The SET command must be the first command in the command structure.
- Since there is only one SCAN command, DSNINDEX is required, and must be specified before any action command.
- SELECT commands are mandatory, while EXCLUDE commands are optional.
- Since there is only one ACTION command, RECOVER is required and must be specified for each SCAN command.

SET Command

Use the SET command to define environment values that will be in effect for the duration of this execution only. Two rules apply to the SET command:

1. It must be specified before any other other command
2. Can not specify more than one

As with all SAMS:Disk SYSIN commands, it can be commented out (nullified) by placing an asterisk “*” in column 1 of the line containing the command.

SYNTAX

The syntax of the SET command is as follows:

```
SET  MODE= , GEN= , VTOCMSG= , MSGLEVEL=
```

MODE=

Specify SIMULATE or SIM to have SAMS:Disk simulate the execution. In simulate mode, SAMS:Disk produces messages and reports as if the supplied action command had really taken place. Specify LIVE, LIV, or remove the SET command entirely to have SAMS:Disk execute the supplied action command. The default value is LIVE.

GEN=

There are 2 source locations containing data sets names that are available to the SET command:

1. Data set names and/or patterns specified by the TABLE= parameter on the SELECT command.
2. Data set names selected by the VTOCS parameter on the SELECT command.

By specifying GEN=YES, SAMS:Disk will print the combined list of data set names to CMDPRINT. By specifying NO, the list of data set names will not be printed. The default value is YES.

Note: The combined list of data set names is the actual candidate list of data sets, not just the data sets that will be processed. Specifically, those data sets not selected for processing will still be listed.

VTOCMSG=

When the VTOCS parameter is specified on the SELECT command, you can optionally have SAMS:Disk generate a message for data sets that are not processed.

By specifying VTOCMSG=YES, message 3822 is issued for each data set not recovered. By specifying NO, this message is not issued. The default value is NO.

MSGLEVEL=

This parameter controls the degree (a range from 0 to 9) of debugging desired, recording the output in MSGPRINT. For instance, a value of '6' will produce a message showing why each data set was or was not selected. A value of '9' will print all selection control blocks available. The default value is 0, which disables the feature.

PASSWORD Command (For VSAM Only)

This optional command is used to provide SAMS:Disk with any password necessary to carry out the functions requested in the command stream. The command should precede the SCAN command that it is intended to cover. For clarity, we recommend that you specify this command just after the SET command, if any, and before all other commands.

SYNTAX

The syntax of the PASSWORD command is as follows:

```
PASSWORD MASTER=, CATALOG=
```

MASTER=

This optional parameter is used to supply SAMS:Disk with the proper level of password it will need to carry out its requested functions. You may supply any of the following passwords:

- catalog master password
- cluster master password
- SAMS:Disk master password

CATALOG=

This optional parameter is used to supply SAMS:Disk with the catalog name to which restored clusters are to be cataloged.

SCAN Command

The SCAN command is used to tell SAMS:Disk to scan the DSNINDEX subfile of the Files data set. A SCAN command must precede any action command. Any SELECT and EXCLUDE commands must follow the SCAN command and precede the action command.

SYNTAX

The syntax of the SCAN command is as follows:

```
SCAN DSNINDEX
```

DSNINDEX

This REQUIRED parameter causes SAMS:Disk to scan the records located in the DSNINDEX subfile.

SELECT Command

This required command allows a user to process only those data sets that satisfy the specified selection test(s).

If a SELECT statement is specified, in order for a data set to be selected, ALL tests specified on a single SELECT command must be met. If only one parameter is specified, only that single test must be met for selection to take place.

For example, if the DSNAMES= parameter or the VOLUMES= parameter, but not both, are specified on a single command, conceptually think “*process only these data sets*” or “*process only these volumes*”. If both the DSNAMES= and VOLUMES= parameters are specified, conceptually think “*process only these data sets if residing on one of these volumes*”.

If multiple SELECT commands are specified for an action command, this has the effect of a logical “OR” between SELECT commands. That is, if any SELECT command is true, the data set is processed, regardless of whether other SELECT commands test true or not.

SYNTAX

The syntax of the SELECT command is as follows:

```
SELECT DSNAMES=, TABLE=, VOLUMES=, TDSN=, CATALOGS=,
      VTOCS, CRITERIA=
```

DSNAMES=

You may specify up to 30 fully qualified names, 30 SAMS:Disk pattern names, or a combination of both. If specified, only those data sets matching any of the names or patterns will be candidates for recovery.

This parameter has two default values.

1. If parameter CATALOGS= is not present on the SELECT command, the default is 'DSNAMES=/', meaning "all" data sets.
2. If parameter CATALOGS= is present on the SELECT command, the default of the DSNAMES= parameter is 'null', meaning "no" data sets.

Note: Both DSNAMES= and CATALOGS= parameters can be specified on the same SELECT command.

CATALOGS=

You may specify a list of up to 30 catalog names, 30 SAMS:Disk pattern names, or a combination of both. If specified, only those catalogs matching any of these names or patterns will be candidates for recovery. If CATALOGS= is not specified,

the default value is 'null', meaning "no" catalogs will be recovered. For related information, turn to the DSNAMES= parameter description on page 85.

TABLE=

You may specify a name of a member in PARMLIB containing the list (table) of fully qualified data set names or patterns. If specified, only those data sets matching any of the names or patterns in the table will be candidates for recovery. The format of the table is described under *"Selection Lists"* on page 455 of the *Systems Guide*.

VOLUMES=

You may specify a list of up to 50 archive volume serial numbers, 50 SAMS:Disk pattern names, or a combination of both. If specified, only data sets that were backed up or archived from the specified list will be candidates for recovery. The default value is '/', meaning "all" volumes.

TDSN=

You may specify a list of up to 30 archive volume data set names, 30 SAMS:Disk pattern names, or a combination of both. If specified, only those data sets that were backed up or archived onto an ARCHVOL matching this naming convention will be candidates for recovery.

VTACS

Causes SAMS:Disk to recover data sets based on a VTAC image created during BACKUP. The zero version backup copy of the VTAC is read for all volumes specified on the VOLUME= parameter, and a list is built of candidate data sets.

Note: If you are familiar with the RECOVER PROC, you will notice a significant difference in the way FMS processes this VTAC image. With the RECOVER PROC, all data sets backed up from the volume are recovered, as long as they have corresponding DSNINDEX records. The FMS PROC simply uses the VTAC image as another table of data set names to do selection (or exclusion) upon. It is for this reason that SET VTACMSGs= defaults to NO.

CRITERIA=

You may specify a series of "criteria tests" to limit the selection of data sets or catalogs to be considered as candidates for recovery. The criteria parameter is specified by a minimum of three subparameters as follows:

(operand1 , operator , operand2)

The operands may consist of any combination of the criteria subparameters described on page 87. The maximum number of subparameters allowed on a single SELECT or EXCLUDE command is 256. Because the CRITERIA= parameter is already very restrictive, you may not specify the VTACS parameter on the same SELECT command. You may, however, add the DSNAMES=, TABLE=, VOLUMES=, or TDSN= parameters.

For logical operators, you may specify any of the following:

Table 3-3. Criteria Operators and their Meanings

Operator	Meaning
EQ	equal to
NE	not equal to
LT	less than
LE	less than or equal to
GT	greater than
GE	greater than or equal to

Criteria Statement Operands

For operands, you may specify any of the following:

- Decimal number constants, such as 10 or 1000.
- Date constants in any acceptable SAMS:Disk format, such as 96200, 96.200, 96/200, 19JUL96, 07/19/96, 1996200, 1996.200, 19JUL1996, 1996/200 or 07/19/1996. If SAMS:Disk has trouble differentiating between a digit Julian date constant, such as 96200, and a decimal number constant, it will issue an appropriate message and ask you to use another format.
- Constant values TRUE, FALSE, YES, NO, ON, or OFF. TRUE, YES, and ON are all equivalent. FALSE, NO, and OFF are all equivalent.
- Expressions that use addition (+) or subtraction (-) to combine a constant value with a field name or other constant value. For example, the expression TODAY-30 represents the date 30 days ago.
- Any of the DSCL select criteria field names listed below.

The following list shows the FMS Select Criteria field name, listed alphabetically. Then after a general discussion on dates, each criteria is described in detail.

For your convenience, the page number where the Field Name description can be found is also listed.

Table 3-4. FMS Selection Criteria Field Names

Field Name	Page #	Definition
ALLOBLKS	88	Allocated in blocks
ALLOCYLS	89	Allocated in cylinders
ALLOPRI	89	Primary allocation amount
ALLOTRKS	90	Allocated in tracks
ARCDT	90	Backup/archive date
ARCTIME	90	Backup/archive time
ARCTYPE	90	Type of DSNINDEX record
CREDIT	91	Creation date in format-1 DSCB
DSNKEY	91	ARCHIVE KEY
DSORG	91	Data set organization
EXPDT	92	Expiration date in format-1 DSCB
JOBNAME	92	Jobname in the format-1 DSCB prior to archival
KBYTES	92	Kilobytes allocated to data set
MODDT	92	Last modification date in format-1 DSCB
PARTBKUP	92	Recover partially backed up PO data sets
RECAT	93	Recataloged to SAMS:Disk Pseudo volume
RESDT	93	Restore date in DSNINDEX record
RESTIME	93	Restore time in DSNINDEX record
TODAY	93	Today's date
USEDT	94	Last use date in format-1 DSCB
USERFLD	94	User field information in DSNINDEX record
VERSION	94	Version number of the DSNINDEX record

Field Name	Page #	Definition
VERSIONARC	95	Version number of Archive DSNINDEX record
VERSIONBKP	95	Version number of Backup DSNINDEX record

ALLOBLKS — Allocated in blocks (yes/no)

This criteria allows for the selection of DSNINDEX records that were allocated in blocks at the time the data set was backed up or archived. For VSAM, this selection criteria test is based on the data component information.

For detail information about when this condition is true, turn to the topic "*ALLOBLKS — Allocated in blocks*" on page 55.

For example, to select only those DSNINDEX records that were allocated in blocks at backup time:

```
CRI=(ALLOBLKS,EQ,YES)
```

ALLOCYLS — Allocated in cylinders (yes/no)

This criteria allows for the selection of DSNINDEX records that were allocated in cylinders at the time the data set was backed up or archived. For VSAM, this selection criteria test is based on the data component information.

For detail information about when this condition is true, turn to the topic "*ALLOCYLS — Allocated in cylinders*" on page 55.

For example, to select only those DSNINDEX records that were allocated in cylinders at backup time:

```
CRI=(ALLOCYLS,EQ,YES)
```

ALLOPRI — Primary allocation amount (number)

The value of ALLOPRI is set to the first extent as specified in the DSNINDEX record. Since the actual primary request value is not recorded in the VTOC, the size of the first extent is displayed. The first extent size is calculated from information in the format-1 DSCB. Unless the data set is very large, or the free space on a volume is badly fragmented, the first extent will accurately reflect the primary space. The value must be specified as a numeric value.

For related information, please turn to the sysparm description for *ALLOCPRI* on page 115 of the *Systems Guide*.

For example, to select only those DSNINDEX records that have a primary extent smaller the 15 tracks:

```
CRI=(ALLOPRI,LT,15,AND,ALLOTRKS,EQ,YES)
```

Note: If sysparm PRIALLOC is left at its default value, Recover processing will consolidate the original extents during allocation. This may increase the number of

data sets selected using ALLOPRI. For more information, review the sysparm description for *PRIALLOC* on page 157 of the *Systems Guide*.

ALLOTRKS — Allocated in tracks (yes/no)

This criteria allows for the selection of DSNINDEX records that were allocated in tracks at the time the data set was backed up or archived. For VSAM, this selection criteria test is based on the data component information.

For detail information about when this condition is true, turn to the topic "*ALLOTRKS — Allocated in tracks*" on page 56.

For example, to select only those DSNINDEX records that were allocated in tracks at backup time:

```
CRI=(ALLOTRKS,EQ,YES)
```

ARCDT — BACKUP/ARCHIVE date (date format)

This criteria allows for the selection of DSNINDEX records based upon their *Written Date* stamps. This parameter is functionally equivalent to the DATE= and EDATE= parameters used by the RECOVER PROC. Any valid SAMS:Disk date format is supported, including TODAY. For related information, see the parameter description of ARCTIME on page 90.

Note: For a list of valid dates, turn to the topic "*Date Formats*" on page 8.

For example, to select only those DSNINDEX records that were written less than a week ago:

```
CRI=(ARCDT,LT,TODAY-7)
```

ARCTIME — BACKUP/ARCHIVE time (time format)

This criteria allows for the selection of DSNINDEX records based upon their *Written Time* stamps. This parameter is functionally equivalent to the TIME= and ETIME= parameters used by the RECOVER PROC.

For example, to select DSNINDEX records that were written on January 22, 1996 between the hours of 11AM and 11PM:

```
CRI=(ARCDT,EQ,96022,AND,ARCTIME,GT,1100,AND,ARCTIME,LT,2300)
```

ARCTYPE — Type of DSNINDEX record

This criteria allows for the selection of DSNINDEX records based upon whether they are an archive or backup copy of the data set. Valid operands are: BACKUP or ARCHIVE.

For example, to select only those DSNINDEX records that were created as a result of executing the BACKUP command:

```
CRI=(ARCTYPE,EQ,BACKUP)
```

Caution: Using this example without any other criteria parameters could potentially "down-level" a data set. This will happen if version zero of the DSNINDEX record being recovered was created by the ARCHIVE command.

For related information, turn to the criteria description of *VERSION* on page 94.

CREDIT — Creation date (date format)

This criteria allows for the selection of DSNINDEX records based upon the Format-1 DSCB creation date of the data set. Any valid SAMS:Disk date format is supported, including TODAY.

Note: For a list of valid dates, turn to the topic "Date Formats" on page 8.

For example, to select only those DSNINDEX records of data sets created in the last 3 months:

```
CRI=( CREDIT,GT,TODAY-90 )
```

DSNKEY — ARCHIVE KEY (number)

This criteria allows for the selection of DSNINDEX records based upon their ARCHIVE KEY. This value must be enclosed in < >, and can represent a tape, cartridge, or disk archive. For an example of an ARCHIVE KEY, refer to the LISTD report on page 295.

For example, to select those DSNINDEX records that have DISK12 as their ARCHIVE KEY:

```
CRI=( DSNKEY,EQ,<DISK12> )
```

DSORG — Data Set Organization

This criteria allows for the selection of DSNINDEX records based upon the value in its DSORG field. Valid values are:

Table 3-5. DSORG Criteria Field Value - FMS PROC

Value	Definition
PS	Physical Sequential
PO	Partitioned
POE	Partitioned Extended
DA	Direct Access
AM	Access Method (VSAM)

For example, to select only those DSNINDEX records that have AM in their DSORG field:

```
CRI=( DSORG,EQ,AM)
```

Note: This example is functionally equivalent to specifying VSAMONLY from the RECOVER PROC.

EXPDT — Expiration date (date format)

This criteria allows for the selection of DSNINDEX records based upon the Format-1 DSCB expiration date of the data set. Any valid SAMS:Disk date format is supported, including TODAY.

Note: For a list of valid dates, turn to the topic *"Date Formats"* on page 8.

For example, to exclude only those DSNINDEX records that contain data sets which have Format-1 DSCB expiration dates between January 1 and July 1, 1996:

```
CRI=( EXPDT,GT,1/1/96,AND,EXPDT,LT,7/1/96 )
```

JOBNAME — JOBNAME in the Format-1 prior to archival

This criteria allows for the selection of DSNINDEX records based on the name of the job in the Format-1 DSCB of the data set prior to being archived. Depending of the OPEN SVC option chosen, this would either be the creating, modifying, or last using JOBNAME, or an account number. For information on OPEN SVC options, see the topic *"Installing the SAMS:Disk SVC"* on page 33 of the *Installation Guide*.

For example, to select only those DSNINDEX records that contain SB15BKUP in their JOBNAME field:

```
CRI=( JOBNAME,EQ,<SB15BKUP> )
```

KBYTES — Kilobytes allocated to the data set (number)

This criteria allows for the selection of DSNINDEX records that contained nnnnn kilobytes at the time the data set was backed up or archived.

For example, to select only those DSNINDEX records that contain data sets that were larger than 50000 kilobytes at backup time:

```
CRI=( KBYTES,GT,50000 )
```

MODDT — Last Modification date (date format)

This criteria allows for the selection of DSNINDEX records based upon the Format-1 DSCB last modification date of the data set. Any valid SAMS:Disk date format is supported, including TODAY.

Note: For a list of valid dates, turn to the topic *"Date Formats"* on page 8.

For example, to select only those DSNINDEX records that contain data sets last modified on May 2, 1993:

```
CRI=( MODDT,EQ,1993123 )
```

PARTBKUP — Recover Partially Backed Up PO's (yes/no)

Default processing excludes partially backed up PO data sets from being selected for recovery. Instead, FMS selects the latest non-partially backed up PO. The objective of this criteria is to recover partially backed up PO's.

For example, to include partially backed up PO's during recover, add the following criteria statement to your command stream:

```
CRI=( PARTBKUP , EQ , YES )
```

RECAT — Pseudo-Volume used when Archived (yes/no)

This criteria allows for the selection of DSNINDEX records based on whether the data set was recataloged to the pseudo-volume at archive time. Valid values are: YES or NO.

For example, to select only those DSNINDEX records that contain data sets cataloged to ARCIVE:

```
CRI=( RECAT , EQ , YES )
```

RESDT — Last RESTORE Date (date format)

This criteria allows for the selection of DSNINDEX records based upon the *Last Restore Date* of the DSNINDEX record. This criteria can be used to restart a failed RECOVER job by placing it on an EXCLUDE statement in conjunction with the VERSION criteria. Any valid SAMS:Disk date format is supported, including TODAY.

Note: For a list of valid dates, turn to the topic "*Date Formats*" on page 8, and for other related information, turn to the criteria description of *VERSION* on page 94.

For example, the typical method of restarting a RECOVER that failed:

```
EXCLUDE CRI=( RESDT , EQ , TODAY , OR , VERSION , NE , 0 )
```

Caution: Be aware that VERSION should always be used in conjunction with RESDT. This is because ALL versions (even back level copies) are candidates to FMS unless specifically excluded or disallowed by selection criteria.

RESTIME — Last RESTORE Time (time format)

This criteria allows for the selection of DSNINDEX records based upon the *Last Restore Time* of the DSNINDEX record.

Note: For related information, turn to the criteria description of *VERSION* on page 94.

For example, to select only those DSNINDEX records that were restored yesterday between 8AM and 5PM:

```
CRI=( RESTIME , GT , 0800 , AND , RESTIME , LT , 1700 , AND , RESDT , EQ , TODAY-1 )
```

TODAY — Today's date (date format)

This value is derived from the system clock. Comparison dates used in conjunction with this criteria may be entered in any acceptable SAMS:Disk format.

Note: For a list of valid dates, turn to the topic "*Date Formats*" on page 8.

USEDT — Last Use date (date format)

This criteria allows for the selection of DSNINDEX records based upon the Format-1 DSCB last use date of the data set. Any valid SAMS:Disk date format is supported, including TODAY.

Note: For a list of valid dates, turn to the topic "*Date Formats*" on page 8.

For example, to select only those DSNINDEX records that contain data sets which were last used on November 1, 1994:

```
CRI=(USEDT,EQ,11/01/94)
```

USERFLD — User Field Information

This criteria allows for the selection of DSNINDEX records based upon the information contained within the 25 byte user field. The comparison is made against the left justified field, and blanks and non-printable Hex characters are not allowed. For related information, see the DSNINDEX record format on page 423, and the DINDEX User Exit on page 215, both in the *Systems Guide*.

For example, to select only those DSNINDEX records that contain Acct#18753 in the User Field:

```
CRI=(USERFLD,EQ,<Acct#18753>)
```

VERSION — Version # of the DSNINDEX record

This criteria allows for the selection of DSNINDEX records based upon the relative version number of the data set. Both archive and backup copies are considered when calculating the version number, and it can be used to restart a failed RECOVER job. For details and an example, please review the *RESDT* criteria on page 93. Valid values are: Negative numeric or zero—the most current version.

For example, to select the latest backup of a data set that has not been archived:

```
CRI=(VERSION,EQ,0,AND,ARCTYPE,EQ,BACKUP)
```

Or, in a recover restart situation, consider the following statement:

```
EXCLUDE CRI=(RESDT,EQ,TODAY,AND,VERSION,NE,0)
```

Note: In the second example, there are two important items to point out:

1. The value of RESDT is very important, especially if the recover began before midnight, and abended after midnight. If this is the case, the appropriate date would be "GE,TODAY-1".

2. IF VERSION was not specified as shown, there is a possibility of "down-leveling" a data set by mistake by selecting a non-zero version of the DSNINDEX record.

VERSIONARC — Version # of ARCHIVE DSNINDEX records

This criteria allows for the selection of DSNINDEX records based upon the relative version number of the data set. Only archive copies are considered when calculating the version number.

Caution: Since this criteria automatically excludes backup versions, you could potentially "down-level" a data set that has a more current backup version in the archives.

Valid values are: Negative numeric or zero—the most current version.

For example, to select the -1 version of an archived copy, ignoring all backup copies, even a more current one:

```
CRI=(VERSIONARC,EQ,-1)
```

VERSIONBKP — Version # of BACKUP DSNINDEX records

This criteria allows for the selection of DSNINDEX records based upon the version number of the data set. Only backup copies are considered when calculating the version number.

Caution: Since this criteria automatically excludes archive versions, you could potentially "down-level" a data set that has a more current archive version in the archives.

Valid values are: Negative numeric or zero—the most current version.

For example, to select the third oldest backup version, ignoring all archive copies, even a more current one:

```
CRI=(VERSIONBKP,EQ,-2)
```

FMS Select Criteria Boolean (AND/OR) Logic

The Boolean logic support of the CRITERIA parameter is invoked by specifying either an AND or an OR between conditional tests. Remember that a conditional test consists of an operand, an operator, and a second operand. You may specify a maximum of 256 conditional tests on a single CRITERIA parameter when each is separated by a compound condition.

For several examples of valid Boolean tests, please turn to page 67.

EXCLUDE Command

The EXCLUDE command allows you to exclude data sets from processing, and it takes precedence over SELECT commands. For instance, if the data set being processed is excluded by any EXCLUDE command, the SELECT of the data set will not occur.

In order for a data set to be excluded, all tests specified on a single EXCLUDE command must be met. If only one parameter is specified, only that single test must be met for the exclusion to be honored. If the EXCLUDE command is specified more than once for any given ACTION command, it has the effect of a logical “OR” between EXCLUDE commands. That is, if any EXCLUDE command is true, then the data set is bypassed, regardless if other EXCLUDE commands test true or not.

For example, if the DSNAMES= parameter or the VOLUMES= parameter, but not both, are specified, conceptually think *“don’t process these data sets”* or *“don’t process these volumes”*. If both the DSNAMES= and VOLUMES= parameters are specified on a single EXCLUDE command, conceptually think *“don’t process these data sets if they reside on these volumes”*.

All parameters that may be specified on the SELECT command are valid parameters for the EXCLUDE command. They operate in exactly the same fashion and with the same defaults.

SYNTAX

The syntax of the EXCLUDE command is as follows:

```
EXCLUDE DSNAMES=, TABLE=, VOLUMES=, TDSN=, CATALOGS=,  
        VTOCS, CRITERIA=
```

DSNAMES=

For a detailed description of this parameter, please refer to the DSNAMES= explanation on page 85.

TABLE=

For a detailed description of this parameter, please refer to the TABLE= explanation on page 86.

VOLUMES=

For a detailed description of this parameter, please refer to the VOLUMES= explanation on page 86.

TDSN=

For a detailed description of this parameter, please refer to the TDSN= explanation on page 86.

CATALOGS=

For a detailed description of this parameter, please refer to the CATALOGS= explanation on page 85.

VTOCS

For a detailed description of this parameter, please refer to the VTOCS explanation on page 86.

CRITERIA=

For a detailed description of this parameter, please refer to the CRITERIA= explanation on page 86.

ACTION Commands

The command RECOVER is known as an action command because it causes SAMS:Disk to take a specific action against a selected data set or volume. This command is required for the FMS proc to function properly.

The RECOVER command and its parameters are described beginning on page 243.

Chapter 4. Reports

The objective of the SAMS:Disk reporting facilities is to provide a rich, comprehensive palette of commands for efficient, common reporting on mixtures of non-VSAM data sets and VSAM clusters.

Within a single execution you may report on non-VSAM data sets, individual VSAM components, DMF/EF Type-2 catalogs, and VSAM clusters.

You may do this based on: scans of VTOCs of disk volumes, scans of DMF/EF Type-2 or VSAM (but not OS CVOL) catalogs, or the results of a catalog superlocate.

You also have a rich set of criteria to select data sets for and exclude data sets from reporting as desired, and a rich set of reports to let you control exactly what you want to see.

Table 4-1. DSCL Commands Available for Reports

Commands Available	JCL Proc Executed	Reports Available
EXCLUDE	DMS	
FIND	DMS	
PASSWORD	DMS	
REPORT	DMS	CLDATA, CIARPT, DSINDEX, DSUTIL, MAPPER, MEMBER, MVDICT, PDSTAT, POCCHH, PODUMP, POTTRD, POXREF, PO1REF, PO2REF, SVDICT
VREPORT	DMS	ALLOCS, ATTRBS, DISTRS, FREESP, LAYOUT
SCAN	DMS	
SELECT	DMS	
SELECT	SMFRPT	SMFRPT
SET	DMS	

General Information

Although managing data storage requires many different tasks to be performed, the ability to generate meaningful reports probably remains as the most common of them. Therefore you should become familiar with the wide range of reporting facilities that SAMS:Disk provides.

First there is the set of “standard reports”, which display data fields proven through the years to be highly relevant. These are “fixed format reports”, that is, reports in which the contents and format of the output have been predefined. They display data obtained from volume VTOC entries and catalog entries, and are commonly produced via batch execution. When combined with the select and exclude capabilities of the command language (DSCL), you have a high degree of flexibility in obtaining just the information you need.

A common complaint in trying to manage VSAM data sets is that even the report information is “completely different”. To help you in this regard, three of the “standard reports” MVDICT, SVDICT and MAPPER produced for non-VSAM data sets from VTOC information are also produced for VSAM clusters from information in the catalog. The catalog-based reports are formatted identically to the VTOC-based reports, however, the results may differ for multivolume data sets. VTOC-based reports will show only the volumes specified in the selection criteria, whereas the catalog-based reports will show all volumes the data set resides on even if the primary volume is the only volume selected. Additional reports CLDATA and CIARPT are also provided that analyze and display information which is indeed unique to VSAM.

Five of the “standard reports”, ALLOCS, ATTRBS, DISTRs, FREESP, and LAYOUT, are considered volume-level reports. These reports do not perform data set level processing. By eliminating this overhead, performance for these reports is greatly improved.

The material presented in the remainder of this section applies to the standard reports obtained via batch.

Requests for report data may be made in two ways: by naming specific data sets (explicit requests) or by selecting certain data sets while scanning the VTOCs or catalogs (implicit requests). The implicit reporting function makes use of the DSCL commands to scan volumes and select or exclude the appropriate data sets. Explicit requests use the FIND command, which is also presented in this section.

Multiple reports may be generated during a single scan of a set of volumes. Records for each type of desired report are generated during the scan phase, are sorted into the appropriate sequence, and then printed by the report writer component of SAMS:Disk.

Performance notes:

1. If you are not interested in VSAM cluster information and are not running any concurrent BACKUP, ARCHIVE, DELETE, RELEASE, or BILLING commands, you can dramatically improve execution time by using the COMPONENTS parameter on the REPORT command. This is due to bypassing catalog overhead for simple REPORT COMPONENTS processing.
2. As described in the Data Storage Command Language section, only one scan of the VTOCs is made, regardless of the number of DSCL commands entered.

REPORT Condition Codes

Condition codes issued while executing the REPORT or VREPORT commands are described in the topic "*Condition Codes*" on page 24.

Restrictions

Volumes to be scanned and reported on must be online.

REPORT Commands and Parameters

The following JCL statements and DSCL commands are used to produce all of the SAMS:Disk reports. For details on DSCL, turn to page 23.

```
//jobname JOB (acct,info),etc.
//STEP EXEC DMS
SCAN REALVOLS
SCAN CATALOGS=
  FIND DSNNAME=, CATALOG=, PASSWORD=
  SELECT VOL=, DSN=, TABLE=, CRITERIA=, VTOCS, CATALOGS=
  EXCLUDE VOL=, DSN=, TABLE=, CRITERIA=, VTOCS, CATALOGS=
  REPORT CLDATA, CLDOPT=, CIARPT, CIAOPT=,
    DSINDX, DSUTIL, MAPPER,
    MEMBER, MVDICT, PDSTAT, POCCHH, PODUMP, POTTRD,
    POXREF, PO1REF, PO2REF, SVDICT,
    CMDGENDSN=, COMPONENTS, EXCLUDE,
    FLAGDAYS=, MEMBERS=, MODULES=,
    NVSAMRPT=, RPT=, SELECT=
  VREPORT ALLOCS, ATTRBS, DISTRs, FREESP, LAYOUT
```

Figure 4-1. DSCL Report Sample

REPORT Command Names

The names of the reports available to the REPORT command shown in the above figure are described below. The additional parameters that can be included on the REPORT command are described beginning on page 106.

CLDATA — Cluster Data Analysis Report

Generate a tailored report displaying the information associated with each cluster. Several different groups of items may be selected for display. A detailed description of the contents of each group selected follows later in this section.

Hitachi does not reliably maintain the information in the catalog entry describing an DMF/EF Type-2 catalog as a cluster itself. Therefore the usage information report lines for an DMF/EF Type-2 catalog cluster itself are also unreliable.

CLDOPT=

- This optional parameter may be used to override the default group selection list used by the Cluster Data Report (CLDATA). It will be ignored by all other report types specified on the REPORT command. Its value is a string of code letters indicating the detail groups to be shown. The code letters follow immediately after the CLDOPT= parameter and are not separated by commas. The order in which the code letters are supplied on the command determines the order of the detail groups on the report. If this parameter is omitted when a CLD report is requested, sysparm RPTCLSEL will be used to determine the default groups to be printed. Following is a list of the valid codes. If an

invalid code is supplied, an error message will be issued and execution will be terminated.

Table 4-2. CLDOPT Values and Descriptions

Value	Description
D	Data component detail
I	Index component detail
E	Extent data
S	Space allocation values
U	Use statistics (default)
P	Protection levels
A	Attributes (non-defaults in use)
Z	Attributes (defaults in use)
R	Recovery volumes
C	Catalog name where cluster is defined
B	Print a blank line
X	Print data set's associations

CIARPT — Cluster Internal Analysis Report

Generate an in-depth analysis of the physical characteristics of data in VSAM clusters. It is composed of three sections:

- catalog information
- detailed graphic analysis
- summary statistics

A detailed description of the contents of each follows later in this section.

CIA reports are not created for SVSAM clusters, XDM/RD database clusters, or DMF/EF Type-2 catalogs as data sets themselves. Hitachi does not maintain all fields necessary to analyze them accurately for this report.

Generating a CIARPT requires CONTROL access to VSAM clusters, not READ access. SAMS:Disk must read the clusters on a Control Interval level.

CIAOPT=(CATLG,DETAIL,DUMP)

- With this optional parameter you can specify options for printing the Cluster Internal Analysis report (assuming you specified the CIA

report). The CATLG keyword instructs SAMS:Disk to print the catalog information for each data set. The DETAIL keyword produces a detailed analysis graph of each used control area in the cluster. This option should be used with prudence because it can produce a large amount of output (each control interval in the data set will occupy a single print line). The DUMP keyword causes a CI Dump to be produced. The order in which you specify these keywords is insignificant.

- If this parameter is NOT specified, default values are retrieved from sysparms VSCIACAT and VSCIADET. However, if the parameter is specified, it overrides the sysparm defaults.

DSINDX— Data Set Index Report

Generate a three-column listing of data set names and the volume on which each resides.

Note: Total allocated tracks in the three volume summary reports (ALLOCS, ATTRBS, and DISTRs) include the VTOC itself and the volume label track. The total allocated tracks for the data set level reports (such as the MVDICT or SVDICT reports) do not include the VTOC or label tracks, but are the sum of the data sets listed on the report.

DSUTIL— Data Set Utilization Report

Generate a listing that includes the special fields as maintained by the SAMS:Disk SVC regarding the usage of each data set. The special fields are:

- Date of last use
- Date of last modification
- Dumber of times the data set has been used

SAMS:Disk assumes a default location in the format-1 DSCB for each of the above fields. If some of these fields are not being maintained, or if they are at offsets different from the defaults, the following six system parameters must be specified to indicate their presence and location. Enter a displacement of zeros for any field not being maintained.

Table 4-3. DSUTIL SYSPARM Displacement Values

SYSPARM	Default Displacement
DSCBLUSD	075 (the Hitachi-defined location)
DSCBLMOD	070
DSCBOPCD	073

By default, the data sets are listed in sequence by date of last use, and the catalog status is displayed. Three system parameters permit slight variations in these op-

tions and are described briefly here. For more complete information, refer to their sysparm descriptions beginning on page 136 of the *Systems Guide*.

Table 4-4. DSUTIL SYSPARM Displacement Variations

SYSPARM	Usage
DSUTILSQ	Sequence by data set name
DSUTILCF	Sequence by last use within index

MAPPER—Volume Map Report

Generate a multiple-column map report for a volume with data set name or other appropriate identifier for each extent type. The CCHH value for each extent is displayed, along with the relative track number, length in tracks and, where appropriate, the number of the extent.

MEMBER— Partitioned Data Set Member Directory Report

Generate a multiple-column listing of members of the partitioned data set in alphabetic sequence. By use of the prefix notation, a listing of subsets of a PDS can be obtained.

MVDICT— Multiple Volume Data Set Dictionary Report

Generate a report of data set attributes listed in alphabetic sequence without regard to volume. Allocation data is included for each data set.

By default, for VSAM clusters, the MVD and SVD reports display each cluster's Control Interval size in the BLKSZ field. If you desire to have the MVD and SVD reports display each cluster's true blocksize instead, specify SAMS:Disk sysparm VSUSEBLK with a value of Y.

PDSTAT— Partitioned Data Set Status Report

Generate a report of one line per data set, which includes allocation and utilization information about member space, directory blocks and directory entries.

POCCHH— Partitioned Data Set Directory (CCHHR Sequence)

Generate a formatted dump of the directory entries for partitioned data set members. The member entries are listed in their beginning CCHHR sequence. (Useful for matching system I/O error messages to specific PDS member names, or for use with performance analysis being done on a PDS.)

PODUMP— Partitioned Data Set Directory (Member Sequence)

Generate a formatted dump of the directory entries for partitioned data set members. The member entries are listed in member name sequence.

POTTRD— Partitioned Data Set Directory (TTR Sequence)

Generate a formatted dump of the directory entries for partitioned data set members. The member entries are listed in their beginning TTR sequence. (Useful for performance analysis of a PDS, or for matching known TTR values back to the actual member/alias names.)

POXREF— DSN Cross-Reference for PDS Member Names

Generate a list in member name sequence of all the member names found in the partitioned data sets that were processed. Listed with each member name are the one or more data sets that contain a member by that name.

PO1REF— DSN Cross-Reference for Member in Only One PDS

Similar to the POXREF report except that the member is listed only when it is found in just one data set. Its suggested use is to find discrepancies between two libraries that should have a perfect member-for- member correspondence, such as a source and its matching load library.

PO2REF— DSN Cross-Reference for Member in Two or More PDS Data Sets

Similar to the POXREF report except that the member is listed only when it is found in two or more data sets. Its suggested use is to identify those members that appear in more than one library, but are intended to be unique to a single library. Examples would be for checking system proclibs (JCL procedure libraries), or for monitoring test libraries versus their production counterparts.

SVDICT— Single Volume Data Set Dictionary Report

Generate a report identical to the MVDICT report, except that the data sets are displayed alphabetically by device type, and then by volume name.

By default, for VSAM clusters, the MVD and SVD reports display each cluster's Control Interval size in the BLKSZ field. If you desire to have the MVD and SVD reports display each cluster's true blocksize instead, specify SAMS:Disk sysparm VSUSEBLK with a value of Y.

Report Parameters

There are several additional parameters that can be included on any REPORT command. These parameters are described below.

CMDGENDSN=

This optional parameter can be specified to cause SAMS:Disk to save the data set names selected for reporting. This is done by writing the name of each selected data set (and the volume where it resides) to the data set specified on this parameter. The data set must be defined as sequential and have a logical record length of at least 50. Any data set that meets all selection criteria and is displayed on the report will also have its name written to this data set. The records in this data set can then be used

as input to generate other commands, whether they be for SAMS:Disk or some other utility.

Note: This facility is available only to VOS3 installations because the dynamic allocation feature of SVC99 is used to allocate the data set.

COMPONENTS

By default, SAMS:Disk reports on non-VSAM data sets and VSAM clusters. Specify this parameter to have SAMS:Disk report on non-VSAM data sets and VSAM components.

If you are not running any BACKUP, ARCHIVE, DELETE, RELEASE, or BILLING commands in the same SCAN/FIND group, you can dramatically improve execution time by specifying this parameter on the REPORT command. This is due to bypassing catalog overhead for simple REPORT COMPONENTS processing.

EXCLUDE

This optional parameter applies only to the MEMBERS parameter. When omitted, a report involving PDS members will include only those corresponding to entries in a member list. Specification of this parameter will exclude listed members.

FLAGDAYS=

Specifying a one- to five-digit value for this parameter will cause an asterisk (*) to appear in the flagdays field on the data set utilization (DSU) report for those data sets that have not been used within the supplied number of days prior to the current date. If the last used date is zero, a “\$” will appear instead, and the creation date will be displayed in the used date field.

MEMBERS=

A list of one or more member names and prefixes may be included when the report to be produced involves the output of PDS members. (A maximum of 20 names is supported.)

MODULES=

A list of one or more modules to be passed control when a non-format-0 DSCB has been retrieved. The parameters passed are the format-4 DSCB for the volume and the non-format-0 DSCB. (A maximum of 20 entries is supported.)

VREPORT Command Names

The names of the reports available to the VREPORT command shown in the figure on page 102 are described below.

ALLOCS — Volume Allocation Summary Report

Generate one report line per volume that provides the total allocated and free space, and the size of the three largest extents. The total number of DSCBs, by type, is also

listed. Refer to “*DISTR*S — Volume Distribution Summary Report” below for additional information.

ATTRBS — Volume Attribute Summary Report

Generate one report line per volume that provides the total number of data sets by data set organization. Totals are also given for device tracks and VTOC DSCBs. Refer to “*DISTR*S — Volume Distribution Summary Report” below for additional information.

DISTRS — Volume Distribution Summary Report

Generate one report line per volume that provides a distribution of data sets based on the number of tracks allocated. The distribution is according to a fixed set of ranges for the size of the data sets. Totals for allocated and free space extents are also included.

Note: Total allocated tracks in the three volume summary reports (ALLOCS, ATTRBS, and DISTR

S) include the VTOC itself and the volume label track. The total allocated tracks for the data set level reports (such as the MVDICT or SVDICT reports) do not include the VTOC or label tracks, but are the sum of the data sets listed on the report.

FREESP — Free Space Detail Report

Generate a report of all free space extents listed in ascending track order for a volume. Starting track and length are listed for each extent.

LAYOUT — Volume Layout Report

Generate a compact list of all extents on a volume. The extents are shown in track sequence, by extent type and length in tracks.

REPORT Field Legend

Fields and indicators used in reports are explained below. Several self-explanatory fields have not been included. The CIARPT and CLDATA report fields are listed separately in the section immediately following.

*	used in DSU report to indicate that a data set has not been used within the period specified on the report command
\$	used in DSU report to indicate that a data set's last used date is zero, so the creation date has been displayed as the last used date
ALLOC	the number of tracks allocated to data sets
AVAIL	the number of free tracks available for allocation
BLKSZ	for non-VSAM, the DCB blocksize of the data set For VSAM, the CISIZE of the data set
CAT	Catalog status. Possible values are:

Table 4-5. Catalog Status Indicators

Status	Meaning
Y	cataloged correctly
N	not cataloged
-	cataloged, but to a different volume

CREDIT	the Julian date the data set was created
CREATING JOB	the first job to open the data set. Note that IEFBR14 allocation jobs will not appear since they merely allocate space and do not open and load any data into the data set.
EXPDT	the Julian date on which the data set expires
FREE	the number of free (format-0) DSCBs in the VTOC
IDLE	the number of unused tracks in a data set
LRECL	logical record size. For VSAM it is the logical record size from the data component.
MODDT	the Julian date the data set was last modified

PRO Data set protection status. Possible values are:

Table 4-6. Data Protection Indicators

Status	Meaning
F	non-VSAM, RACF-indicated
R	for read/write password protection
W	for write password protection
C	VSAM, control-interval password protected
M	VSAM, master password required for access
R	VSAM, read-only password protected
U	VSAM, update access password protected

PRIM primary space allocation value, expressed in the type of units in which the allocation was made. Since the actual primary request value is not recorded in the VTOC, the size of the first extent is displayed. The first extent size is calculated from information in the format-1 DSCB. Unless the data set is very large, or the free space on a volume is badly fragmented, the first extent will accurately reflect the primary space. If you desire to alter this field, review the sysparm description for *ALLOCPRI* on page 115 of the *Systems Guide*.

SEC secondary space allocation value, expressed in the type of units in which the allocation was made.

TYPE the type of allocation unit: CYL, TRK, or BLK

USEDT the Julian date the data set was last used (opened)

CLDATA Description

The Cluster Data Report (CLD) contains information about all clusters selected for reporting. The report is organized as follows:

1. The report is subdivided into groups and where possible a group will be shown on one print line. A user may optionally select to suppress any or all detail groups.
2. Most groups are further subdivided into items that are shown in a “keyword” fashion; that is, the item name followed by an equal sign (=) and then the value (for example, RECSZ=100).
3. VSAM definitions incorporate many default values. In order to produce a concise report, SAMS:Disk may suppress a group item if the default value is in effect. For example, the default attribute for a cluster is NOSPAN, so if NOSPAN is in effect, NOSPAN will not be displayed in the attribute group. However, if SPAN is in effect, SPAN will be displayed, as it is a non-default value.
4. If all items in a group are suppressed, the printing of the entire group will also be suppressed. For example, if all attributes in effect are the default values, the attribute line will be suppressed.
5. Where “non-keyword” values are to be printed, SAMS:Disk lists the values in alphabetical order. This simplifies locating a particular group item.
6. An attempt has been made to shorten the conventional report line length in order to improve the usability of SAMS:Disk in a terminal environment.

Cluster Name Line

The cluster name line shows the name, the cluster type, the primary volume serial, the total tracks allocated, and the percentage of space used. This line is always displayed first and cannot be suppressed. If the cluster resides on multiple volumes, the primary volume for the data component will be displayed under the VOLUME heading, followed by a “+” sign (the remaining volumes on which the cluster resides can be found by looking at the extent detail lines). Also, the total number of control area and control interval splits (data + index) will be shown if splits have occurred. If more than the minimum number of extents are allocated, the number of extents are also shown; that is, more than one for an ESDS or RRDS and more than two for a KSDS.

Note: KSDS clusters defined using the key-range attribute and all VSAM catalogs have at least three extents and hence the number of extents will be shown.

A cluster data report in which all other detail lines have been suppressed provides a concise cluster summary report.

Data Component Detail Group (D)

The data component detail group will always provide the average record size (RECSZ=) associated with the cluster. If the maximum record size differs from the average, the maximum record size (MAX=) will also be shown. The control interval size (CISZ=) for the data component (whether specified by the user or chosen by VSAM) will always be provided.

If any control area or control interval splits have occurred in the data portion, the number of splits will be displayed here. If neither CA nor CI splits have occurred, this item will be suppressed.

The data set name associated with the data component will be printed on the subsequent line if sysparm VSCOMPNM is specified with a value of Y. When overriding this sysparm in job streams that use other than the DMS proc, specify PRINT.SYSPARMS.

If any freespace is defined for a KSDS, the freespace percentages will also be displayed (FREESPACE%-CI= and FREESPACE%-CA=).

Index Component Detail Group (I)

The index component detail line is printed only for KSDS type clusters. The length of the key field (KEYL=) is always provided. The relative position of the key field within the data record is provided if its value is greater than zero (RKP=).

If the index contains more than two levels, the number of index levels (LEVELS=) is provided. The control interval size (CISZ=) for the index component (whether specified by the user or chosen by VSAM) will always be provided.

If any control area or control interval splits have occurred in the index portion, the number of splits will be displayed here. If neither CA nor CI splits have occurred, this item will be suppressed.

The data set name associated with the index component will be printed on the subsequent line if sysparm VSCOMPNM is specified with a value of Y. When overriding this sysparm in the job stream, specify PRINT.SYSPARMS dd.

Extent Detail Line (R)

One extent detail line is printed for each unique extent that is allocated to the cluster. The extent detail line shows the volume serial that contains the extent, the first and last absolute track address of the extent, the total number of tracks allocated to the extent, and the word "INDEX" if the extent contains only the index of a KSDS. Since embedded indexes share some extents with the data portion of the cluster, these common extents will not be duplicated for the index component. This should be kept in mind when the user is comparing an JSCVSUT LISTCAT output to the SAMS:Disk CLDATA report.

The extent limits are shown in cylinder and head format (cccc—hh) and are displayed in hexadecimal. The maximum number of extents that can be displayed is 32.

Space Allocation Detail Line (S)

The allocation detail line shows the amount of space requested at definition time. The data and index component values may have been specified by the user or chosen by VSAM based on the SPACE parameter provided for the cluster or model. The format is similar to the OS/VS SPACE parameter used on a dd JCL statement. The first value indicates the type of allocation (CYL or TRK). Second is the primary or initial quantity requested of the indicated type. Third is the secondary quantity for subsequently extending the amount of allocated space. If no secondary quantity was requested, this value is suppressed.

If the cluster type is keyed, the index allocation is provided in the same format.

Note: When a cluster is defined with a space allocation of records, Access Method Services converts this amount to an equivalent amount of storage in either tracks or cylinders. Therefore, SAMS:Disk will always report allocation amounts in either tracks (TRK) or cylinders (CYL), and never in records.

Use Statistical Detail Group (U)

The statistical detail group shows the number of records subjected to the indicated action. If any value is zero, that item will be suppressed. If all values are zero, the entire statistical detail line will be suppressed. Following is the list of actions and associated definitions:

Table 4-7. Use Statistical Group Definitions

Action	Definition
ADD	number of records added since completion of the initial load
DEL	number of records deleted
EXCP	count the number of EXCPs issued against this cluster
GET	number of times any record was retrieved
TOT	total number of user data records currently in the cluster
UPD	number of records updated

Note: These statistics pertain to the data component only.

LAST MOD DATE and TIME for each cluster is shown also.

Password Detail Group (P)

The password detail group shows either the levels of protection or the actual passwords required for access at each level. If the proper password has been supplied (either the catalog's master password, the SAMS:Disk master password or the cluster's password), the actual passwords are provided at the following levels:

Table 4-8. Password Group Protection Levels

Value	Definition
M	Password for master level access
C	Password for control interval access
U	Password for update or write access
R	Password for read only access

If the proper password has not been supplied, each password will be replaced by the level of access as follows:

Table 4-9. Password Replacement Values

Value	Definition
M	Master
C	Control
U	Update
R	Read

If access is not restricted at a particular level, that level will be suppressed. If access is not restricted at any level, the entire password detail line will be suppressed. If the cluster is RACF-protected, a literal of "RACF" will be displayed on the password report line.

If the number of attempts allowed to supply the entity's password is other than zero, the literal ATTEMPTS= will also appear on the password detail line. Beside the literal will be the actual number of attempts that are allowed.

Attribute Detail Group (A,Z)

The non-default attributes that are in effect for the cluster can be displayed by selecting the (A) group.

The default attributes will be displayed if the (Z) group is specified. Items within each group are listed alphabetically.

The following table lists the non-default and default values that will be displayed:

Table 4-10. Attribute Detail Group Descriptions

Non-Defaults(A)	Defaults(Z)	Description
ERASE	NO	Data secure erase
IMBED	NO	Store seq set w/data
ORDER	NO	Use vols in order
REPL	NO	Repeat index on trk
SHR(-,-)	SHR(1,3)	Share options
SPAN	NO	Allow RECSZ > CISZ
SPEED	RECOVERY	No preformat on load
UNIQUE	SUBALLOCATE	Cluster = space
UNIQUEKEY	NON	AIX with unique keys
WCK	NO	Verify data after write

Catalog Name Line (C)

The Catalog Name report line displays the catalog name in which the cluster is defined.

Blank Line (B)

This SELECT type is used to generate blank line(s) in the Cluster Data Report. It is used primarily to enhance readability. As many blank lines as desired can be printed. Double-spacing can be achieved by supplying two “B” SELECT types consecutively.

Association Lines (X)

Each association for the entity being reported on will be displayed on a separate line. Immediately following the association name on the print line is a literal indicating the type of association (for example, cluster, path, data, etc.).

CIA Report Description

The Cluster Internal Analysis report (CIARPT) contains information about the internal structure of data in VSAM clusters. The report has a special format for each of the three different types of VSAM clusters:

- key-sequenced data sets (KSDS)
- entry-sequenced data sets (ESDS)
- relative record data sets (RRDS)

These reports were designed to allow as much similarity as possible between them, while still allowing for the differences in data structures of each cluster type. The reports can be generated as a function of the REPORT command, or may be produced as a byproduct of the ARCHIVE or BACKUP command. It should be noted that in order to produce these reports, SAMS:Disk must read the clusters with control interval access (which is the method used by default with SAMS:Disk).

Each of the reports have three distinct portions to them:

- information obtained from the catalog (optional)
- a graphic display of each control interval in every control area that contains data (also optional)
- a statistics section that summarizes the key information about the cluster.

The first two sections can be dynamically selected by specifying the CIAOPT= parameter of the REPORT, BACKUP and ARCHIVE commands, or can be controlled by specifying sysparms VSCIACAT and VSCIADET (for detail information, refer to their sysparm descriptions beginning on page 185 in the *Systems Guide*).

CIA Report KSDS Catalog Information Section

DEV TYPE

This field is the device class of the volume(s) on which the component resides. All of the volumes for a particular component must be of the same device type, but the device type can be different for the data and index components.

EXTENTS

The total number of extents in which the component resides.

CI SIZE

The control interval size for the component. This is the unit of information transmitted to or from auxiliary storage by VSAM.

PHY RECSZ

This is the physical record size used to store information in the cluster. There is a whole number of physical records that comprise one control interval.

REC / TRK

This is the number of physical records stored on each track. This will give an indication of the effectiveness of the blocking factor on this particular DASD device.

HI USED RBA

The high used relative byte address (RBA) in this component. This address effectively marks where the first empty control area resides, or the end of the data set if all space has been used.

HI ALLOC RBA

The high allocated relative byte address (RBA) in this component. This address marks the end of the data set in its currently allocated extents.

TRKS / CA

The number of tracks per control area. For most larger clusters this value should be the same as the number of tracks per cylinder on this type of DASD. This would represent the ideal control area size of one cylinder.

CI / CA

The number of control intervals that are contained in one control area. This is an important number in key-sequenced data sets when determining the appropriate control interval size for the index. Refer to the topic "*Unreferenceable Data Control Intervals*" on page 122 for more details.

FREE %-CI

The control interval freespace percentage with which the cluster was defined. This is the percentage of freespace to be reserved in each control interval when it is loaded.

FREE %-CA

This value represents the percentage of control intervals in every control area that are to be left empty when the cluster is loaded. It is specified when the cluster is initially defined.

AVG LRECL

This is the length of the average record (specified when the cluster was defined) that the user expects the data set to contain. When variable-length records occur in a data set, this value may or may not be an accurate reflection of the actual data, but is an estimate supplied by the user by which VSAM estimates space requirements when loading the cluster. The actual average record length of the data set is shown in the AVG RECSZ field in the statistics summary of this report.

MAX LRECL

This is the maximum record length allowed in the cluster. If this field is the same value as AVG LRECL, the file contains fixed-length records; otherwise they are variable-length.

TOTAL RECORDS

The total number of records the catalog expects the data set to contain. This may not reflect the true number of records. The catalog information is accurate only when data sets are properly closed after updating. If a system crash occurs during updating, the catalog statistics may be inaccurate (this value alone in no way affects the integrity of the cluster). Most database systems using VSAM do not perform normal close operations, so these data sets will rarely have the proper value in the catalog. For the true number of records in the data set, refer to the TOTAL RECORDS field in the statistical summary of this report. This number is obtained by actually counting the number of records that reside in the data.

CI SPLITS

This is the total number of control interval splits that have occurred in the data set.

CA SPLITS

This is the total number of control area splits that have occurred in the data set.

Index Component Catalog Information

Most of the catalog information displayed for the index component of the cluster is in the same format as the data fields, with the following additions:

SEQ SET RBA

This field represents the relative byte address of the first control interval in the sequence set of the index.

INX LVLS

This is the number of index levels that occur in the index. For direct access retrieval of records in the data component, this number represents the number of index control intervals that must be accessed to locate the proper control interval in the data component that contains the record.

HI LEVEL RBA

This is the relative byte address of the control interval that contains the highest level segment in the index. For direct access retrieval of records, this is the first index segment that is retrieved to begin the search for the specified key value.

KEY LTH

This is the length of the key in the data portion of the cluster, as specified by the user when the cluster is defined.

RKP

This is the relative key position. It represents the offset from the beginning of the data record to where the key of the record can be found. An RKP of zero indicates the key starts in the first character of the data record.

CIA Report Sequence Set Graphic Analysis

The sequence set analysis graph is produced only for key-sequenced clusters (since they are the only type that has an index). Its purpose is to give a graphic picture of the control intervals that comprise the sequence set portion of the index. Note that this does not include control intervals that are part of the index set, because these are not needed unless the data set is processed by keyed direct access.

This graph contains one print line for each control interval in the index's sequence set. The graph line can consist of four different print characters:

- X's represent header information overhead
- F's represent space used to hold free control interval pointers
- spaces symbolize space which is unused in the control interval
- I's represent space that is used to hold the compressed keys and pointers to the data control intervals.

A legend is printed under the final line of the graph to indicate the use of these characters. We will now take a closer look at each of these values.

Header Information

The header information is always 24 characters in length. This header contains control information about the index record itself. The contents of this information are not important to this discussion, but you should be aware that these 24 characters are not available to hold actual data. You may notice that on different clusters, however, a different number of X's may appear on each line of the graph. This is because of the scaling factor that was used on the graph, which is based on the control interval size of the component (you can see both of these values on the print line that precedes the graph). With a control interval size of 4096 (we will refer to these values as CI size from now on), the scaling factor is 1/64. This makes the 24 bytes insignificant for graphing purposes, and will therefore not show up. On the other hand, a CI size of 512 produces a graphing factor of 1/8, and therefore three X's will be displayed to represent the 24 characters.

Freespace Pointers

The freespace pointers (represented by F's) may or may not exist in the control interval. These pointers are used to point to control intervals in the data component that are not being used in this control area. (Each sequence set CI "owns" one specific control area in the data component.) The length of each of these pointers is governed by a field in the header information. The column at the side of the graph labeled FREE PTRS is the number of free pointers found in the sequence set record. This is an actual count, and not the number of characters needed to hold the pointers. Therefore, this count represents the actual number of control intervals in the

data component's control area that are not being used. We will discuss freespace pointers in more detail in a little while.

Freespace

The freespace in the sequence set (represented by spaces) is quite different from the freespace pointers. This is used to display the space that is unused in the sequence set record. If all of the CIs in the control area for the data are already being used, then this area is of no significance (remember, all of the CIs are being used if the freespace pointers value is zero). However, if there are unused CIs in the data's control area, this area may become much more important. We will discuss this more in the topic *"Unreferenceable Data Control Intervals"* on page 122. For now, just remember that the value under the column FREE SPACE represents the number of empty positions in the sequence set record that can be used for other purposes.

Index Entries

The I's in the graph symbolize space that is used to hold the pointers to actual user data in the data component. This consists of some characters for control information, the pointers to the data control intervals, and most importantly the key value (in compressed format) of the highest record in each control interval. This key is used by VSAM in keyed access to determine if a particular data control interval has the record it is looking for, without having to read each data control interval itself.

There are two column headings that relate directly to the index entries in the graph. The first one is labeled AVG ENTL, which stands for the average entry length of an index entry. This value is the average number of characters required to reference a single data control interval, which includes control information overhead and the compressed key of the data. This value is important when looking at the issue of unreferenceable control intervals, which we will look at in the topic *"Unreferenceable Data Control Intervals"* on page 122.

The second column applicable to index entries is AVG KEYL. This value represents the average number of characters required to hold each compressed key. What this value tells you is how well VSAM is compressing the keys used to reference the data. This information can be used when deciding upon an explicit CI size for the index if you want to reorganize the cluster with SAMS:Disk (using ARCHIVE and RESTORE). The new CI size can be explicitly stated on the RESTORE command.

Description of Sequence Set Graphic Headings

This section will explain each of the values printed next to the graphic display. Those columns that were described in preceding sections will only be summarized at this point.

CONTROLS CA AT RBA

This is the relative byte address of the start of the control area that this index CI controls. This number is used by the VSAM access method to determine where the

data corresponding to the keys in this index starts. This value is provided for informational purposes only.

FREE PTRS

This is the number of free control interval pointers that exist in this particular index control interval. In other words, this number represents the number of data control intervals that are not being used in the corresponding data control area.

USED PTRS

This is the number of pointers being used in this index CI. This corresponds to the number of data control intervals being used in the control area. This value, added to the FREE PTRS value, is the same as the number of control intervals per control area in the data component.

FREE SPACE

This is the number of empty bytes or characters in the index control interval that can be used to hold pointers to the data control area. This value was discussed previously.

AVG ENTL

The average number of bytes required to hold both the control information and compressed data key for each control interval being referenced in this data control area.

AVG KEYL

The average length to which each data key compresses, excluding any system control information.

UNREF CI'S

The number of unreferenceable data control intervals in this control area. This value is explained more fully below. If this value is not zero, DASD space is being wasted. If this number is large, redefine the cluster with different control interval sizes.

CA SEQ

This value represents the relative order of the control area in key sequence. In a cluster that has had no control area splits, this value should always be incremented by one for each line in the graph.

CA DEV

This value represents the relative deviation between the current control area and the one that preceded it in key sequence. In an organized cluster, this value will be 1. Since this value is “normal”, it is suppressed on the report so you will see blanks instead. If the value is other than 1, the control area does not physically reside next to the control area beneath it in key sequence. What does this mean? Assuming a control area size of one cylinder (the optimal value for most clusters), the arm that

reads the data on DASD has to move the number of cylinders shown under the REL DEV column to read the data in keyed sequential mode. If the number printed has a “-” after it, the seek would be backwards. It is important to understand that this number has no relevance when the cluster is accessed in direct key mode, since the index is searched to find where the record resides (such as in online databases). But if the file is accessed in keyed sequential mode—as SAMS:Disk does to produce a backup copy of the data set—then the head movement may have a negative impact on the performance of the read operations.

Unreferenceable Data Control Intervals

To explain what causes data control intervals to become unreferenceable requires an explanation of how a sequence set control interval is structured. Remember that one sequence set control interval governs a single control area in the data component of the VSAM cluster. To see this structure, refer to the diagram below:

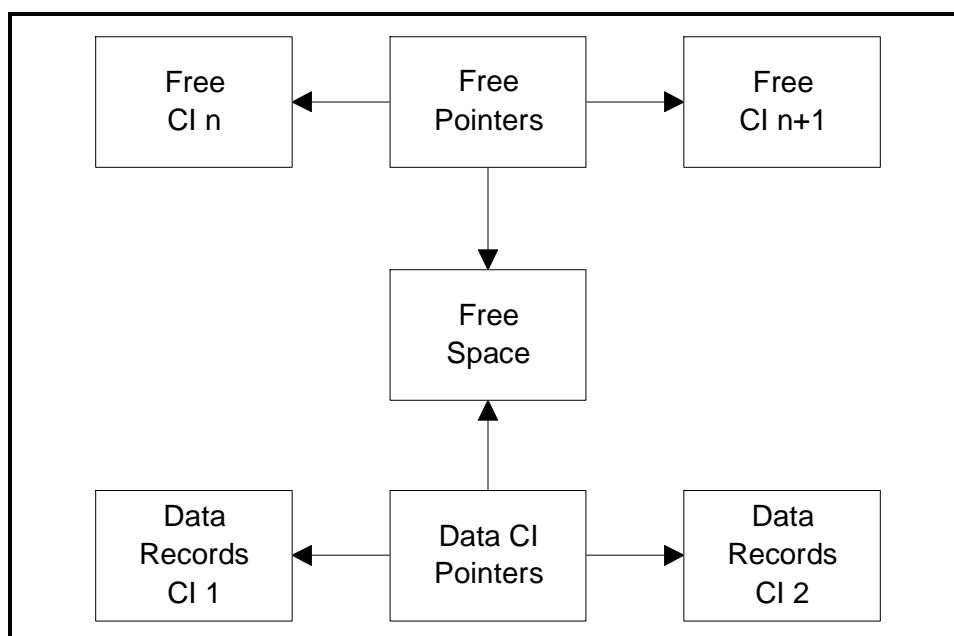


Figure 4-2. Sequence Set Control Interval

Before a data area contains any records, the sequence set record contains only free pointers to the control intervals in the data control area. This lets VSAM know that all of the control intervals are available and ready for use. When VSAM has a request to add a record, it checks to see if there is space available in any of the used control intervals. If it determines that a free control interval must be used to hold the new record, the sequence set is checked to see if there are any available. If there are, the key of the record is compressed and placed in the right side of the sequence set record, along with a pointer to the appropriate data control interval. At the same time, the free pointer is removed from the left side of the sequence set CI. This process repeats itself every time a new control interval is required.

As you can see, each time a free pointer is used, some space is reclaimed from the left side of the sequence set CI, while additional space is used on the right side. The free pointer will always be from one to three characters in length. However, the information that gets added on the right side is almost always larger, since it must contain pointers, compressed keys and control information. So the space in the center of the sequence set CI (labeled free space) slowly gets used up as each new control interval is used in the control area.

Normally this is not a problem, because catalog management tries to not let you define an index control interval size that is too small to reference all of these data control intervals. But there are occasions, however rare, when the keys of the data do not compress as well as expected. When this occurs, there is not enough free space remaining in the sequence set CI to hold the compressed key of the data record. Therefore, even though there are free control intervals in the data area, they cannot be used because the sequence set CI has run out of room to hold the necessary information. This space in the data component effectively becomes dead space.

Obviously this is not a good condition. There are several methods that can be used to rectify this condition, and the best choice depends on the circumstances in your environment. The first alternative is to reorganize your cluster and define it with a larger index control interval size. This will add more free space to the sequence set CIs. Another approach is to increase the data CI size, since this would mean that fewer data control intervals would have to be referenced by the same sized index. The last alternative is to redefine the cluster with different space attributes, such that the control area size would be decreased (refer to the Hitachi publication Access Methods Services for a discussion of how space allocation affects control area size). This last approach would be the least desirable from a performance point of view, unless the control area size of the cluster is already less than one cylinder.

You may have encountered a situation that is caused by this very issue during device conversions of VSAM data sets. For example, say you are moving a cluster from a 8595 to a 8598 and the control area size is one cylinder. You archive the cluster and attempt to do a restore, only to get an error message (JSC0009I-S) during the define that says “control interval size calculation unsolvable”. What occurred is that catalog management failed the define because it is trying to avoid this very situation. Since the control area size is one cylinder, there are now more data control intervals that have to be referenced by the same-sized index CI, and catalog management decided you would end up with unreferenceable data control intervals. So what was valid for the smaller device is not valid for the larger one. The solution to this problem is the same as the one outlined above.

CIA Report Data Component Graphic Analysis

The data component analysis graph provides a picture of what each used control area in the data component looks like. Each control area is represented by a separate box graph. At the conclusion of each control area, summary totals are printed for the appropriate fields in the control area.

This graph contains one print line for each control interval in the control area. Most graph lines will consist of four different print characters:

- D's represent space used to hold data records.
- Spaces symbolize space that is unused in the control interval,.
- *'s represent space that is considered unusable for data storage (dead space).
- X's represent control information overhead.

A legend is printed under the final line of the graph to indicate the use of these characters. In control areas that contain spanned records, you will also see the following symbols:

F	Represents the control interval containing the first segment of a spanned record.
I	Represents any intermediate segments.
L	Represents the last segment of a spanned record, followed by *'s to indicate the dead space in the last control interval.

The legend for spanned records will only appear after those control areas that actually contain at least one spanned record. You may also notice some control intervals with a literal of “— FREE —” inside them. These are control intervals that are marked as free in the sequence set CI and therefore contain no data. We will now take a closer look at each of these values.

Data Records

These characters (D's) are used to represent the space actually used to contain data records. The number of characters in all of the data records in this control interval can be found under the column BYTE TOTAL.

Freespace

This is space that is currently not being used in the control interval and is therefore available either to add new records or to expand existing ones in variable length files. This space is represented by spaces.

Dead Space

This is the amount of free space in a control interval that cannot be used to add other records, so SAMS:Disk categorizes it as dead space. This space is represented by the character *. In fixed-length record files this amount is a constant value and is calculated as follows:

$$CISIZE - (RECSIZE * RECORDS \text{ PER } CI) - 10$$

For variable-length files SAMS:Disk makes an estimation of dead space based upon the records processed to that point. Any free space quantity that is less than the average record size +3 is considered dead space. The +3 is added in to assume the worst case of not more than two consecutive records having the same length records, which would cause three bytes of control information to be added to the control interval. It should be noted that this space is available to expand existing records. Also, even though the space is considered dead now, a control interval split later on would make the area usable again.

The unused space in the last segment of every spanned record is also marked as dead space. It is a VSAM restriction that any control interval that contains a portion of a spanned record cannot contain any other records, even if there is enough free space to hold additional records.

Control Information Overhead

The character X is used to depict the space used to hold control information for the control interval. This contains such information as the length of the data in the control interval and the length of any freespace. For fixed-length files the length of this area is always 10. For variable-length files it is 4 bytes, plus 3 bytes for each data record in the control interval (actually, VSAM does save some space when more than two consecutive records have the same length of data).

Description of Data Component Graphic Headings

REC TOTAL

This is the total number of data records found in the control interval. When the CI contains a portion of a spanned record, the literal SPAN will appear in front of the record count. A value of zero is displayed for the first and any intermediate segments of the spanned record, and a value of one is displayed for the last segment.

BYTE TOTAL

This is the total number of characters found in all of the data records in this control interval.

FREE SPACE

This is the amount of unused space in the control interval. It can be used to add new records or to expand records in a file that can contain variable-length records.

DEAD SPACE

This is the number of bytes that SAMS:Disk considers unusable in the control interval.

OVERHEAD

This is the number of bytes required to hold the system control information.

% USED

This is the percentage of the cluster currently being used. For the purposes of this figure, used space can either be space used to hold data records or space required by VSAM for control information.

% DEAD

This is the percentage of the control interval considered to be unusable. It is calculated by dividing the number of dead bytes by the control interval size.

REL SEQ

This value represents the relative order of each control interval in the control area by key sequence. In a control area that has all of its control intervals in key sequence, this value will always be incremented by one for each line in the graph.

REL DEV

This value represents the relative deviation between the current control interval and the one that preceded it in key sequence. In an organized cluster, this value will be 1. Since this value is “normal” it is suppressed on the report so you will see blanks instead. If the value is other than 1, the control interval does not physically reside next to the control interval beneath it in key sequence. This normally would have no impact on the performance of the cluster. The only exception to this would be when the control area size is less than one cylinder and the deviation would cause the DASD device to have to shuttle between two different cylinders to read the control intervals in keyed-sequence mode.

CIA Report KSDS Data Component

The summary information is the only portion of this report that is not optional. It contains all of the statistics applicable to the data set as a whole. We will now examine each of the fields on this report.

TOTAL RECORDS

The total number of data records in the data set. This may or may not have the same value as the total records value in the catalog section of this report. Remember that the total records value printed in the catalog section is based on the value maintained in the catalog while the number in this section is based on the actual records in the cluster. This value is therefore more accurate.

TOTAL BYTES

This is the total number of bytes or characters of data that are used to hold the data records in the cluster.

SPACE %—USED

The percent used gives an indication of the space already consumed in the data component. It accounts for space used to hold data records, control information overhead, and any dead space encountered in the data set.

USED CI COUNT

This field gives the number of control intervals in the data component that contain data records.

FREE CI COUNT

This is the number of control intervals in the data component that are marked as free by the sequence set in the index. Because these control intervals are marked as free, SAMS:Disk does not attempt to read them.

UNUSED CI COUNT

This is the number of control intervals that do not contain any data records but that are not marked as free in the index. The significance of this is that the control interval does have a compressed key pointing to it from the index, and therefore can only be used by records that have keys that fall into the range of the control interval. This number can be an indication of wasted space in a cluster, particularly in files that have their primary activity occur randomly throughout the cluster. If this number becomes significantly large in a data set, a reorganization may be in order so that the unused control intervals can be reclaimed.

CI %—USED

This value is the percentage of control intervals allocated to the data component that contain at least one data record.

The remaining portion of this print line has several estimated values listed. These values are based upon the data found in the cluster and assume that the same data characteristics will apply to future records added to the data set.

Usable Free Bytes

This is the number of bytes that can be used to store new data records. It does not include space in unreferenceable control intervals, or dead space in control intervals (either from spanned records or unusable space). This should give a true picture of what space is available in which to add records.

Estimated New Record Additions

There are two values that SAMS:Disk may calculate for a cluster, depending on whether or not the cluster contains any spanned records. (The cluster must actually have at least one spanned record, not merely be defined with the SPANNED attribute.) If the cluster has no spanned records, the only value to be calculated will be the number of non-spanned records that can be added. This is based on both free space in used control intervals and also on all unused control intervals. For control intervals that already contain data records, the calculation is based on the free space in the control interval and the average record size of records processed to that point. For unused control intervals, the value is based on how well records fit into the used control intervals.

When a cluster contains spanned records, the calculation becomes more difficult. In effect, SAMS:Disk tries to simulate the addition of both spanned and non-spanned records at the same percentage as that which occurs in the data itself. Without going into the elaborate details, SAMS:Disk basically does the following:

- Any free space in used control intervals is assignable to non-spanned record additions only.
- Spanned records require whole numbers of free control intervals to be allocated, based on the average number of control intervals required to hold a spanned record.
- An iterative process is used by SAMS:Disk that simulates the addition of records throughout the unused portion of the data set at the prescribed percentage.

Because of the method used in simulation, the percentage of new record additions may vary slightly from the actual percentage found in the data.

Estimated Dead Space

This is the number of bytes that SAMS:Disk estimates are not usable for data storage. The value is based on both absolute dead space (such as unused bytes in the last segment of a spanned record or unused space in data sets with fixed-length records) and assumed dead space (in variable-length files, space that is available for record expansion but that is not large enough to contain a data record). Keep in mind that the estimated dead space in a control interval may be reclaimed later by a control interval split.

MIN RECSZ

This is the size of the smallest record that was found in the data set.

MAX RECSZ

This is the size of the largest record that was found in the data set.

AVG RECSZ

This is the average size of each record in the data set. For fixed-length files, MIN, MAX and AVG record sizes will be the same value.

MIN RECS/CI

This value is the minimum number of records that were found in any used control interval.

MAX RECS/CI

This value is the maximum number of records that were found in any used control interval.

AVG RECS/CI

This value is the average number of records that were found in each used control interval.

MIN FSPC/CI

This is the minimum number of unused bytes found in any used control interval. This space may actually be considered dead space in other calculations of this report if SAMS:Disk determines the space cannot be used to hold data records. Control intervals that contain a segment of a spanned record are ignored in all four of the freespace per CI values.

MAX FSPC/CI

This is the maximum number of unused bytes found in any used control interval.

AVG FSPC/CI

This is the average number of unused bytes found in any used control interval.

AVG—% FSPC/CI

This is the average percentage of freespace to used space in all control intervals read by SAMS:Disk.

AVG CI OVHD/REC

This value represents the average number of bytes required for control information for each data record. This information is required by VSAM to access the records but is not part of the user's data. This value can be useful in estimating storage requirements when reorganizing a cluster (add this value to average record size to come up with an accurate value of space required to hold each data record).

MAX CI DEV

This value represents the maximum relative deviation between any two control intervals when processed in key sequence.

AVG CI DEV

This value represents the average relative deviation between control intervals when processed in key sequence. A value of 1.0 represents the "best case" of each control interval in key sequence physically residing next to one another. As this number increases in size, so does the disorder of the data set.

SPANNED RECORD INFORMATION

A separate print line will be printed for all clusters that have spanned records. Because of the poor performance associated with spanned records — both in CPU and DASD utilization — these values should be checked closely. If a fairly large percentage of records are spanned, you should consider increasing the control interval size of the data set.

TOTAL SPAN RECORDS

This is the total number of spanned records that reside in the data set.

% of TOTAL RECS THAT ARE SPANNED

This is the percentage of records in the data set that do span control intervals.

MIN SPAN RECSZ

This is the size of the smallest spanned record in the data set.

MAX SPAN RECSZ

This is the size of the largest spanned record in the data set.

AVG SPAN RECSZ

This is the average size of all spanned records in the data set.

SPAN DEAD SPACE

This is the amount of dead space in the data set attributable to spanned records. In essence, this is the total number of bytes that are not being used in the last control interval of each spanned record.

CIA Report KSDS Index Component Summary

The summary information for the index component of the data set is for the sequence set only. Since SAMS:Disk processes the cluster in key sequence, it does not need any of the records in the index set, which is used to access the cluster in keyed direct access mode.

CI TOTAL IN SEQ SET

This is the total number of control intervals used to hold the index's sequence set. This also equates to the number of control areas being used in the data set, since each sequence set record controls a single data control area.

FULL CI TOTAL

This is the number of control intervals in the sequence set that have no free control interval pointers to the data component. In other words, this is the number of sequence set records that point to control areas in which all of the data control intervals are being used.

UNREF CI TOTAL

This is the total number of data control intervals that are currently — or are projected to be in the future — unreferenceable because the index's control interval size is too small. For related information, see the topic *"Unreferenceable Data Control Intervals"* on page 122.

FREE CI TOTAL

This is the number of control intervals in the index component that are currently unused.

MINIMUM FSPC/CI

This is the minimum amount of free (unused) space found in any sequence set control interval.

MAXIMUM FSPC/CI

This is the maximum amount of free (unused) space found in any sequence set control interval.

AVERAGE FSPC/CI

This is the average amount of free (unused) space found in all sequence set control intervals.

AVERAGE COMP KEYL

This is the average number of bytes to which each key in the sequence set compressed. This value will give an idea as to how effective the compression algorithm is. It is of most value when looking at what control interval size to use for the index. If the keys are compressing well and the MINIMUM FSPC/CI value is sufficiently high, you may want to consider reorganizing the cluster and specifying a smaller control interval size for the index.

AVG INX ENTRY LTH

This value is similar to the AVERAGE COMP KEYL, except it includes the overhead information required to store keys in the sequence set. Therefore, if you subtract the AVERAGE COMP KEYL from this value, you will have the average number of bytes of control information overhead that is required for each index entry.

MAX DATA CA DEV

This value represents the maximum relative deviation that occurred between two consecutive data control areas in key sequence. In a completely organized cluster with no control area splits, this value will be 1. Refer to the description of CA DEV in the graphic analysis section for a more detailed explanation of this value.

AVG DATA CA DEV

This is the average relative deviation of all control areas in the data component. Again, the ideal value is 1.0. As the number gets higher, greater seek distances are required to process the cluster in key-sequenced mode. This value is not relevant to direct key retrieval, since the sequence set is not used. Also, if the data set is on a high activity volume, this value may be meaningless, since head movement will probably occur between control area reads to satisfy requests for other data on the pack.

CIA Report ESDS Catalog Information Section

This section of the report is identical to the catalog section described for KSDS clusters, with the exception of four fields that are not applicable:

- Free %-CI
- Free %-CA
- CI splits
- CA splits

Please refer to the description provided for KSDS clusters for a description of the catalog fields.

CIA Report ESDS Graphic Analysis

As with the catalog information, the graphic analysis section for ESDS clusters is very similar to the data component analysis for KSDS clusters. Since the graphs themselves are identical, we don't need to go over them again, but the column totals are slightly different.

REC TOTAL

This is the total number of data records found in the control interval. When the CI contains a portion of a spanned record, the literal SPAN will appear in front of the record count. A value of zero is displayed for the first and any intermediate segments of the spanned record, and a value of one is displayed for the last segment.

BYTE TOTAL

This is the total number of characters found in all of the data records in this control interval.

DEAD SPACE

This is the number of bytes that SAMS:Disk considers to be unusable in the control interval. Since ESDS clusters can only have records added to the end of the data set, any unused space in control intervals prior to the last one is dead space (that's why there's no freespace column on this graph). You may see some ESDS clusters with a lot of dead space shown for them. These data sets are preformatted with dummy records (for various reasons) and do not look identical to "pure" ESDS clusters. Because of this, SAMS:Disk cannot give an accurate picture of free and dead space in the last control area of the data set.

OVERHEAD

This is the number of bytes required to hold the system control information.

% DEAD

This is the percentage of the control interval considered to be unusable. It is calculated by dividing the number of dead bytes by the control interval size.

CIA Report ESDS Data Component Summary

As with the preceding sections of the ESDS report, this section is basically a subset of fields described for the KSDS data component. Please refer to that section of the manual for a description of the appropriate fields.

CIA Report RRDS Catalog Information Section

The catalog information for relative record data sets is also similar to KSDS and ESDS clusters. The two new fields are LRECL and SLOTS/CI. Since RRDS clusters must have fixed-length records by definition, LRECL gives the length of each record “slot” in the data set. The field SLOTS/CI tells how many record slots fit in each control interval.

CIA Report RRDS Graphic Analysis

The graph provided for relative record clusters is identical to the preceding cluster types in format, with the exception that spanned records are not supported by VSAM. You will therefore never see a second legend line under the graph box.

REC TOTAL

This is the total number of used or “active” slots (records) in the control interval.

BYTE TOTAL

This is the total number of characters found in all of the active slots in this control interval. This is a simple calculation of used slots times the logical record length.

EMPTY SLOTS

This is the number of slots that are currently marked as empty in the control interval.

CIA Report RRDS Data Component Summary

The summary information for relative record data sets is relatively straightforward, especially compared to the previous cluster types.

TOTAL RECORDS

The total number of data records in the data set. This is the number of used or “active” slots that contain user data.

TOTAL BYTES

This is the total number of bytes or characters contained in the data records of the cluster.

TOTAL SLOTS IN DATA SET

This is the total number of slots that exist in the data set. It combines the total used and the total empty.

SLOTS %—USED

The percent used tells what percentage of the slots in the cluster are already being used.

FREE SLOTS PAST HIGH RRN

This number tells you how many empty slots exist beyond the last slot that is being used.

USED CI COUNT

This is the number of control intervals that occur prior to and including the last control interval that contains a data record.

UNUSED CI COUNT

This is the number of control intervals between the last control interval that contains a data record and the end of the data set (high allocated RBA).

CI %—USED

This value is the percentage of control intervals in the data set that occur prior to and including the last control interval that contains an active record.

SPACE %—DEAD

Unlike the other types of clusters, this value is a constant value for each data set. It is dependent on control interval size and the number of slots per control interval. The amount of dead space in any RRDS’s control intervals can be calculated as follows:

$$(\text{CI size} - 4) - ((\text{LRECL} + 3) * \text{Slots per CI})$$

Analyzing the CIA Report

Obviously there is a lot of information provided in these reports, especially the report for key-sequenced data sets. It's probably even more obvious that the information isn't much good unless it can be interpreted and put to good use. Unfortunately, it is impossible to develop absolute guidelines for interpreting all of this information and applying it universally to all data sets. We can, however, point out some things to look for in these reports if you are trying to tune your VSAM environment. We will begin with the simplest report (RRDS) and proceed to the ESDS and KSDS reports.

The one performance item that all three of these reports have in common is the EXTENTS value in the catalog information section. This will tell you how many extents the cluster occupies. A high value will not only impair performance when processing the data set, but it also leads to fragmented storage volumes. If possible, clusters should be archived and restored to consolidate the extents into one contiguous area.

Analyzing RRDS Relative Record Data Sets

There's really not much that can be done to tune RRDS clusters. Other than checking the extent count, the only other value of interest is the dead space percentage printed in the summary section. If an inefficient control interval size is used, an inordinate amount of space would be wasted. Using the following formula for dead space, try substituting different control interval sizes to see which one produces the least dead space:

$$(\text{CI size} - 4) - ((\text{LRECL} + 3) * \text{Slots per CI})$$

Keep in mind the type of processing that is normally used on the data set. For direct processing, it is generally more efficient to use smaller control intervals, while sequential processing is more efficient with larger control interval sizes. Even though a particular CI size may not use DASD as efficiently as another, it may be a better choice if the data set is used frequently.

Analyzing ESDS Entry-Sequenced Data Sets

As with the other types of VSAM clusters, entry-sequenced data sets generally perform better with larger control interval sizes, both in terms of read/write performance and DASD utilization. As with the other types also, this will vary depending on the primary access mode used to process the data set — either direct or sequential. Since not many applications process entry-sequenced data sets in direct mode by RBA address, the larger CI size will probably be more effective. Your installation may have requirements limiting the CI size to decrease the system resources required for VSAM buffers. If this is the case, don't change CI size without checking first.

A very important set of values is printed if the data set is defined with the SPANNED attribute and contains at least one spanned record. Because of the tre-

mendous overhead involved with spanned records (both performance and DASD), they should definitely be the exception and not the rule in a cluster. Check the percentage of spanned records in the data set. Depending on the cluster, even 1 percent might be too high for performance reasons. Check the minimum, maximum, and average spanned record size. It may be possible to redefine the cluster with the next-larger control interval size and eliminate the spanned records altogether.

Analyzing KSDS Key-Sequenced Data Sets

Key-sequenced data sets are the hardest of the cluster types to tune properly because of all the variables involved. Since the parameters governing tuning are installation- and data set- dependent, it would be impractical (if not impossible) to list an exhaustive set of criteria to follow. We will try to list the more important values to check.

- As with the other cluster types, check the number of extents the data set occupies. Consolidate extents with an archive and restore to reduce volume fragmentation. (Archiving the data set to disk and then restoring it will avoid operator intervention to mount archive tapes!)
- As with entry-sequenced data sets, see if the data set has any spanned records. If it does, consider increasing the CI size of the data component if practical. If the cluster is accessed primarily in an online environment, the spanned records may still be a proper choice because of the extra overhead involved in transferring larger control intervals.
- Check the index's control interval size. Many clusters get defined with a CI size that is too large for the index. If the CI size is greater than 512, verify that the index is indeed the proper size. This can be done in part by looking at the minimum freespace per CI value and at the average index entry length found in the index's statistics summary section. If the minimum freespace is less than the difference between the current and the next-lower valid CI size (valid CI sizes for the index are 512, 1024, 2048, and 4096), you are already at the lowest CI size. If this is not the case, multiply the average index entry length by the number of control intervals per control area in the data component (field CI/CA in the catalog information section). This will give you a rough idea of the space needed for indexing the average data control area. There is one important point to keep in mind: if you make the CI size too small, you may force VSAM to more index levels, which is undesirable in an online, direct access environment. There is an entire section devoted to optimizing VSAM's performance in Hitachi's *VSAM Programmer's Guide*, which discusses the impact of control interval size on performance. Before doing any tuning on clusters, this entire section should be read thoroughly.
- Compare the CI size to the physical block size. The optimal situation is for the two to have the same value. If not, a different control interval

size may be in order. Consult the VSAM Programmer's Guide for a table showing the correspondence of CI size to physical block size for space utilization guidelines (in the section on optimizing VSAM's performance).

- For clusters used extensively in sequential access mode (as opposed to direct key), check the average control area deviation value. In a data set with a large number of CA splits, the seek movement can contribute significant overhead. You may want to consider reorganizing the cluster, and maybe also increasing the freespace percentages at restore time to reduce the frequency of splits in the future.
- Look at the unreferenceable CI count. If it is other than zero, you may be wasting DASD space. It is impossible to tell from just the statistics summary if the space is currently unusable or if it is a projection of future problems. Remember that SAMS:Disk simulates the process that VSAM uses when allocating new control intervals from the freespace pointers. Therefore, there may indeed be enough freespace for current usage, but SAMS:Disk is predicting a problem with future additions. If you want to know if the problem exists currently, look at the sequence set graph that is produced. Under the UNREF CIs heading you will see the sequence set records that have unreferenceable CIs. If the FREESPACE value for the control interval is less than the AVG ENTL, the space problem exists right now.
- Look at the dead space percentage. If a high percentage of your data set contains dead space, you should consider going to a higher data control interval size. As a rule, DASD utilization is more efficient at the higher CI sizes. Also, if spanned records are present in the data set, they may be contributing significantly to the problem. Again, consider increasing the CI size.
- Look at the space usage percentage and estimated record additions values. For data sets that are continually expanding, these numbers will give an indication that a reorganization with additional space may be in order. This may keep your data set from going into additional extents or, worse yet, running out of space.

Utilization By Date and Time Report

To obtain this report, execute a four-step job that:

- Selects abbreviated data from SMS type 14 and 15 records
- Sorts the selected data into data set name sequence
- Converts start/stop date and time into grids
- Writes the report

Note: The SMS report requires its input from a sequential data set. If your SMS records reside in a VSAM cluster, the supplied SMFRPT procedure should be modified to add an Hitachi JSCVSUT utility step at the beginning to REPRO the records into a sequential data set to be passed to SAMS:Disk. Verify that this has been done before executing the report for the first time.

SMS Report JCL

Only one SELECT command per execution is allowed.

```
//SMSDATA EXEC SMFRPT
//SYSIN      DD  *
              SELECT COUNT= ,DSN= ,START= ,STOP= ,CLOSE ,TAPE
```

Select Command and Parameters

COUNT=

The maximum number of SMS records to be read in this execution, specified as a 1- to 10-digit number between 1 and 2147483647. SMS records past this number will be bypassed. If not specified, this parameter will default to 10000000.

DSN=

A list of one or more data set names or patterns for which a report is desired. Omission of the parameter will result in SMS records for all data set names being selected. (A maximum of 50 entries is supported.)

START=

The Julian date for which reporting is to begin. Records with a start date earlier than this date will be bypassed.

STOP=

The Julian date for which reporting is to terminate. Records with a close date later than this date will be bypassed.

CLOSE

The presence of this simple parameter will cause the start date to be set equal to the stop (SMS CLOSE) date. Otherwise, the start date is obtained from the reader time stamp field in the SMS type 14 or 15 record.

TAPE

The presence of this parameter will result in tape data sets being included in the report. The volume serial number field is be filled with blanks for all tape records.

The generation and version numbers will be reset to zero for all GDG data sets.

An example of the command is provided for a case in which reporting is desired for all data set names beginning with VOTE and MIS and for the individual data set PAYROLL.MASTER. Also, the presence of the CLOSE parameter will cause only one entry to be placed in the day and hour grids for each data set activity record, since start time will equal stop time.

```
//SMSDATA EXEC SMFRPT  
          SELECT DSN=(VOTE/,MIS/,PAYROLL.MASTER),CLOSE
```

Sample DSCL REPORTS

The remainder of this chapter displays a sample SYSPRINT of each of the DSCL REPORT commands that are available.

ALLOCS

1996.228 AUG 15, 1996				VOLUME ALLOCATION SUMMARY												PAGE			
THURSDAY 1.10 PM																SAMS:DISK 9.1.0			
				----DATA SET CONTROL BLOCKS (DSCB) COUNTS----							-----TRACKS-----				-----FREE SPACE-----				
VOLUME	TYPE	ADDR	STAT	FREE	F1	F2	F3	F4	F5	F6	DEVICE	ALLOC	AVAIL	IDLE	EXTENTS	THREE LARGEST			

WRK000	8598	0340	NONE	764	83		2	1	1		26550	2250	24300	473	18	10380	5565	348	

TOTAL VOLS				1	764	83	2	1	1						18				

TOTAL TRACKS PER DEVICE											26550								

TOTAL TRACKS ALLOCATED												2250							
AVAILABLE												24300							
IDLE												473							

Figure 4-3. Sample Volume Allocation Summary Report

ATTRBS

1996.228		AUG 15, 1996				VOLUME ATTRIBUTE SUMMARY										PAGE	
THURSDAY		1.10 PM														SAMS:DISK 9.1.0	
DEVICE		UNIT	SMS	MOUNT	ALLOCATION	-----DATA SET ORGANIZATION-----						--TOTAL--	---DSCBS---		---TRACKS---		
VOLUME	TYPE	ADDR	STAT	STATUS	ATTRIBUTE	PS	DA	PO	IS	AM	UN	DATA SETS	VTOC	ALLOC	DEVICE	ALLO	
-----		-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	
WRK000	8598	0340	NONE	RESIDENT	PUBLIC	38	2	18		25		83	901	87	26550	225	
TOTAL VOLS		1	TOTALS			38	2	18		25		83					
TOTAL DSCBS VTOC													901				
ALLOC														87			
TOTAL TRACKS PER DEVICE															26550		
ALLOC															225		

Figure 4-4. Sample Volume Attribute Summary Report

CIARPT - ESDS Cluster

1987.338	DEC 04, 1987	C L U S T E R I N T E R N A L A N A L Y S I S										PAGE	1
FRIDAY	8.13 AM											SAMS:Disk	8.2
CLUSTER NAME: LABI.TJP.VERIFY.CLUSTER (ESDS)													

+++++													
=====+ DATA COMPONENT CATALOG INFORMATION +=====													
+++++													
DEV TYPE	EXTENTS	TOTAL TRACKS	CI SIZE	PHY RECSZ	REC/ TRK	HI USED RBA	HI ALLOC RBA	TRKS /CA	CI/ CA	AVG LRECL	MAX LRECL	TOTAL RECORDS	
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	
3380	1	1	16384	4096	10	16384	32768	1	2	50	100	12	
+++++													
=====+ DATA COMPONENT CONTROL INTERVAL ANALYSIS +=====													
+++++													
RBA VALUE	DATA CONTROL AREA					GRAPH SCALE=1/256	CISZ=16384	REC TOTAL	BYTE TOTAL	DEAD SPACE	OVER HEAD	% DEAD	

0	DDDD							12	960	0	10	.0	
16384	-- FREE --							0	0	0	0	.0	

LEGEND - 'D'=DATA ' '=FREESPACE '**'=DEADSPACE 'X'=CONTROL INFO OVERHEAD								12		0		0.0	
									960		10		
+++++													
=====+ DATA COMPONENT STATISTICAL INFORMATION +=====													
+++++													
-----ESTIMATED-----													
TOTAL RECORDS	TOTAL BYTES	SPACE %-USED	USED CI COUNT	UNUSED CI COUNT	CI %-USED	USABLE FREE BYTES	--NEW RECORD ADDITIONS-- NON-SPANNED	SPANNED	DEAD SPACE	SPACE %-DEAD			
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----		
12	960	2.9%	1	1	50.0%	31,798	505	0	0	.0%			
MIN RECSZ	MAX RECSZ	AVG RECSZ	MIN RECS/CI	MAX RECS/CI	AVG RECS/CI	AVG CI OVHD/REC							
-----	-----	-----	-----	-----	-----	-----	-----						
80	80	80.0	12	12	12.0	.8							

Figure 4-6. Sample VSAM ESDS CIARPT (1 of 6)

1987.338 DEC 04, 1987
FRIDAY 8.13 AM

CLUSTER INTERNAL ANALYSIS

PAGE 2
SAMS:Disk 8.2

CLUSTER NAME: LABI.TJP.SPAN.TEST (KSDS)

RBA VALUE	DATA CONTROL AREA (CONTINUED)	REC TOTAL	BYTE TOTAL	FREE SPACE	DEAD SPACE	OVER HEAD	% USED	% DEAD	REL SEQ	REL DEV
3072	FFX SPAN-0	502	0	0	10	100.0	.0	6		
3584	IIIX SPAN-0	502	0	0	10	100.0	.0	7		
4096	LLXXXX SPAN-1	196	0	306	10	40.2	59.7	8		
4608	DDDX SPAN-1	2	160	342	0	10	33.2	.0	12	2-
5120	FFX SPAN-0	502	0	0	10	100.0	.0	10	6	
5632	LLXXXX SPAN-1	340	0	162	10	68.3	31.6	11		
6144	FFX SPAN-0	502	0	0	10	100.0	.0	13	3	
6656	IIIX SPAN-0	502	0	0	10	100.0	.0	14		
7168	LLXXXX SPAN-1	196	0	306	10	40.2	59.7	15		
7680	-- FREE --	0	0	512	0	0	.0	.0		
8192	-- FREE --	0	0	512	0	0	.0	.0		
8704	-- FREE --	0	0	512	0	0	.0	.0		

*** INTERMEDIATE LINES NOT SHOWN ON SAMPLE REPORT ***

93184	-- FREE --	0	0	512	0	0	.0	.0		
93696	-- FREE --	0	0	512	0	0	.0	.0		

LEGEND - 'D'=DATA ' '=FREESPACE '*'=DEADSPACE 'X'=CONTROL INFO OVERHEAD 13 87,936 147 5.4 1.2
SPANNED RECORDS: 'F'=FIRST SEG 'I'=INTERMEDIATE SEG 'L'=LAST SEG 4,942 1,183

+++++
=====+ DATA COMPONENT STATISTICAL INFORMATION +=====

TOTAL RECORDS	TOTAL BYTES	SPACE %-USED	USED COUNT	CI FREE COUNT	CI UNUSED COUNT	CI %-USED	USABLE FREE BYTES	--NEW RECORD NON-SPANNED	ADDITIONS SPANNED	DEAD SPACE	SPACE %-DEAD
13	4,942	6.6%	15	169	0	8.1%	87,936	140	69	1,183	15.4%

Sample DSCL REPORTS

1987.338		DEC 04, 1987		C L U S T E R I N T E R N A L A N A L Y S I S								PAGE		5	
FRIDAY		8.13 AM										SAMS:Disk		8.2	
CLUSTER NAME: LABI.TJP.SPAN.TEST (KSDS)															

MIN	MAX	AVG	MIN	MAX	AVG	MIN	MAX	AVG	AVG-%	AVG CI	MAX CI	AVG CI			
RECSZ	RECSZ	RECSZ	RECS/CI	RECS/CI	RECS/CI	FSPC/CI	FSPC/CI	FSPC/CI	FSPC/CI	OVHD/REC	DEV	DEV			
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----			
60	500	122.2	1	3	1.8	5	445	282.6	55.1%	5.2	6	2.0			
-----SPANNED RECORD INFORMATION-----															
TOTAL SPAN % OF TOTAL RECS			MIN SPAN		MAX SPAN	AVG SPAN		SPAN DEAD							
RECORDS THAT ARE SPANNED			RECSZ		RECSZ	RECSZ		SPACE							
-----			-----		-----	-----		-----							
4			30.7%		600	1,200		960.5		1,178					
+-----+ INDEX COMPONENT STATISTICAL INFORMATION (SEQUENCE SET ONLY) +-----+															
+-----+															
CI TOTAL	FULL CI	UNREF CI	FREE CI	MINIMUM		MAXIMUM	AVERAGE		AVERAGE	AVG INX	MAX DATA	AVG DATA			
IN SEQ SET	TOTAL	TOTAL	TOTAL	FSPC/CI		FSPC/CI	FSPC/CI		COMP KEYL	ENTRY LTH	CA DEV	CA DEV			
-----	-----	-----	-----	-----		-----	-----		-----	-----	-----	-----			
1	0	0	17	1,792		1,792	1,792.0		0.7	3.7	1	1.0			

(Page 3 of 6)

1987.338 DEC 04, 1987
FRIDAY 8.13 AM

CLUSTER INTERNAL ANALYSIS

PAGE 6
SAMS:Disk 8.2

CLUSTER NAME: M134TAA.CSI (KSDS)

+ DATA COMPONENT CATALOG INFORMATION +																	
+ DATA COMPONENT CATALOG INFORMATION +																	
DEV	TYPE	EXTENTS	TOTAL TRACKS	CI SIZE	PHY RECSZ	REC/ TRK	HI USED RBA	HI ALLOC RBA	TRKS /CA	CI/ CA	FREE %-CI	FREE %-CA	AVG LRECL	MAX LRECL	TOTAL RECORDS	CI SPLITS	CA SPLITS
3350		1	1,350	4096	4096	4	8552448	21381120	30	116	10	5	23	142	227,291	2	0

+ INDEX COMPONENT CATALOG INFORMATION +													
DEV	TYPE	EXTENTS	TOTAL TRACKS	CI SIZE	PHY RECSZ	REC/ TRK	HI USED RBA	HI ALLOC RBA	SEQ SET RBA	INX LVLS	HI LEVEL RBA	KEY LTH	RKP
3350		1	30	1024	1024	15	479232	506880	460800	2		0	23 0

+ SEQUENCE SET CONTROL INTERVAL ANALYSIS +														

RBA VALUE	INDEX SEQUENCE SET	GRAPH SCALE=1/16	CISZ= 1024	CONTROLS CA	FREE USED FREE	AVG	AVG	UNREF	CA	CA				
				AT RBA	PTRS PTRS SPACE	ENTL	KEYL	CI'S	SEQ	DEV				
460800	XX	-----	-----	0	6 110	5	8.9	5.7	6	1				
461824	XXF	-----	-----	475136	14 102	15	9.4	6.2	13	2				
462848	XXF	-----	-----	950272	9 107	11	9.0	5.9	8	3				
463872	XXF	-----	-----	1425408	19 97	11	9.9	6.7	18	4				
464896	XXF	-----	-----	1900544	15 101	13	9.5	6.3	14	5				
465920	XX	-----	-----	2375680	5 111	13	8.7	5.6	4	6				
466944	XXF	-----	-----	2850816	14 102	15	9.4	6.2	13	7				
467968	XX	-----	-----	3325952	7 109	19	8.8	5.6	5	8				
468992	XXF	-----	-----	3801088	12 104	12	9.3	6.1	11	9				
470016	XXFF	-----	-----	4276224	29 87	12	10.9	7.7	28	10				
471040	XXFF	-----	-----	4751360	29 87	18	10.8	7.6	28	11				
472064	XXF	-----	-----	5226496	19 97	12	9.9	6.7	18	12				
473088	XX	-----	-----	5701632	5 111	16	8.7	5.5	3	13				
474112	XXFF	-----	-----	6176768	29 87	16	10.8	7.7	28	14				
475136	XXFF	-----	-----	6651904	26 90	19	10.5	7.3	24	15				
476160	XX	-----	-----	7127040	5 111	306	6.1	2.9	0	16				
477184	XXFF	-----	-----	7602176	33 83	14	11.3	8.2	32	17				
478208	XXFFFF	-----	-----	8077312	73 43	424	11.5	8.3	33	18				
LEGEND - 'X'=HEADER 'F'=FREE CI ENTRIES ' '=FREE SPACE 'I'=INDEX ENTRIES				349	951	286								
				1,739										

Sample DSCL REPORTS

1987.338 DEC 04, 1987
FRIDAY 8.13 AM

CLUSTER INTERNAL ANALYSIS

PAGE 8
SAMS:Disk 8.2

CLUSTER NAME: M134TAA.CSI (KSDS)

```

=====+ DATA COMPONENT CONTROL INTERVAL ANALYSIS +=====

```

[illegible]

1987.338 DEC 04, 1987
FRIDAY 8.13 AM

CLUSTER INTERNAL ANALYSIS

PAGE 10
SAMS:Disk 8.2

CLUSTER NAME: M134TAA.CSI (KSDS)

RBA VALUE	DATA CONTROL AREA (CONTINUED)	REC TOTAL	BYTE TOTAL	FREE SPACE	DEAD SPACE	OVER HEAD	%	%	REL SEQ	REL DEV
389120	DD	XXXXX	142	3336	435	0	325	89.3	.0	97
393216	DD	XXXXXX	142	3338	433	0	325	89.4	.0	98
397312	DD	XXXXXX	142	3336	435	0	325	89.3	.0	99
401408	DD	XXXXXX	142	3338	433	0	325	89.4	.0	100
405504	DD	XXXXXX	142	3336	435	0	325	89.3	.0	101
409600	DD	XXXXXX	142	3338	433	0	325	89.4	.0	102
413696	DD	XXXXXX	142	3336	435	0	325	89.3	.0	103
417792	DD	XXXXXX	142	3338	433	0	325	89.4	.0	104
421888	DD	XXXXXX	142	3340	425	0	331	89.6	.0	105
425984	DD	XXXXXX	137	3259	422	0	415	89.6	.0	106
430080	DD	XXXXXX	137	3259	428	0	409	89.5	.0	107
434176	DD	XXXXXX	138	3282	411	0	403	89.9	.0	108
438272	DD	XXXXXX	137	3259	425	0	412	89.6	.0	109
442368	DD	XXXXXX	138	3278	421	0	397	89.7	.0	110
446464	DD	XXX	72	1692	2235	0	169	45.4	.0	74 37
450560	-- FREE --		0	0	4096	0	0	.0	.0	
454656	-- FREE --		0	0	4096	0	0	.0	.0	
458752	-- FREE --		0	0	4096	0	0	.0	.0	
462848	-- FREE --		0	0	4096	0	0	.0	.0	
466944	-- FREE --		0	0	4096	0	0	.0	.0	
471040	-- FREE --		0	0	4096	0	0	.0	.0	
LEGEND - 'D'=DATA ' '=FREESPACE '*'=DEADSPACE 'X'=CONTROL INFO OVERHEAD		15,475	74,990	35,906	84.2	0.0				

+-----+ DATA COMPONENT STATISTICAL INFORMATION +-----+												
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+												
-----ESTIMATED-----												
TOTAL RECORDS	TOTAL BYTES	SPACE %-USED	USED CI COUNT	FREE CI COUNT	UNUSED CI COUNT	CI %-USED	USABLE FREE BYTES	--NEW RECORD NON-SPANNED	ADDITIONS-- SPANNED	DEAD SPACE	SPACE %-DEAD	
227,291	5,984,602	35.2%	1,739	349	3,132	33.3%	13,839,995	517,706	0	1,171,456	16.4%	
MIN RECSZ	MAX RECSZ	AVG RECSZ	MIN RECS/CI	MAX RECS/CI	AVG RECS/CI	MIN FSPC/CI	MAX FSPC/CI	AVG FSPC/CI	AVG-% FSPC/CI	AVG CI OVHD/REC	MAX CI DEV	AVG CI DEV
23	103	26.3	19	159	130.7	62	2,235	433.1	10.5%	1.6	37	1.0

+-----+ INDEX COMPONENT STATISTICAL INFORMATION (SEQUENCE SET ONLY) +-----+												
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+												
CI TOTAL IN SEQ SET	FULL CI TOTAL	UNREF CI TOTAL	FREE CI TOTAL	MINIMUM FSPC/CI	MAXIMUM FSPC/CI	AVERAGE FSPC/CI	AVERAGE COMP	AVG INX ENTRY	AVG INX LTH	MAX DATA CA DEV	AVG DATA CA DEV	
18	0	286	27	5	424	52.8	6.3	9.5		1	1.0	

CIARPT - RRDS Cluster

1987.338	DEC 04, 1987	C L U S T E R I N T E R N A L A N A L Y S I S										PAGE	1
FRIDAY	8.13 AM											SAMS:Disk	8.2
CLUSTER NAME: LABI.TJP.RRDS.CLUSTER (RRDS)													

+++++													
=====+ DATA COMPONENT CATALOG INFORMATION +=====													
+++++													
DEV TYPE	EXTENTS	TOTAL TRACKS	CI SIZE	PHY RECSZ	REC/ TRK	HI USED RBA	HI ALLOC RBA	TRKS /CA	CI/ CA	LRECL	SLOTS /CI	TOTAL RECORDS	

3380	1	2	12288	4096	10	36864	73728	1	3	80	148	444	
+++++													
=====+ DATA COMPONENT CONTROL INTERVAL ANALYSIS +=====													
+++++													
RBA VALUE	DATA CONTROL AREA					GRAPH SCALE=1/192	CISZ=12288	REC TOTAL	BYTE TOTAL	EMPTY SLOTS			

0	D							3	240	145			
12288	-- FREE --							0	0	148			
24576	-- FREE --							0	0	148			

LEGEND - 'D'=DATA ' '=FREESPACE '**'=DEADSPACE 'X'=CONTROL INFO OVERHEAD								3	240	441			
+++++													
=====+ DATA COMPONENT STATISTICAL INFORMATION +=====													
+++++													
TOTAL RECORDS	TOTAL BYTES	TOTAL SLOTS IN DATASET	SLOTS %-USED	FREE SLOTS PAST HIGH RRN	USED CI COUNT	UNUSED CI COUNT	CI %-USED	SPACE %-DEAD					

3	240	888	.3%	885	1	5	16.6%	.0%					

Figure 4-7. Sample VSAM RRDS CIARPT

CLDATA

1987.341	DEC 07, 1987	C L U S T E R D E T A I L A N A L Y S I S				PAGE	1
MONDAY	6.18 PM					SAMS:Disk	8.2
				TRKS	PERCENT		
COMPONENT	TYPE	VOLUME	ALLOC	USED	SPLITS / EXTENTS		

LABJ.MJB.TESTKD4	KSDS	LABS56	60	53			
DATA - RECSZ=215 CISZ=16384 FREESPACE%-CI=10 FREESPACE%-CA=10							
INDEX - KEYL=8 CISZ=512							
EXTENT -	1. LABS56	0054--00	0054--1D	TRKS=30			
EXTENT -	2. LABS56	0055--00	0055--1D	TRKS=30	INDEX		
ALLOC - DATA=(CYL,(1,1)) INDEX=(CYL,(1,1))							
STATS - EXCP=37 GET=400 TOT=200							
STATS - LAST MOD=NOV 19, 1987 1:50 PM LAST USED=NOV 15, 1987							
ATTR - IMBED, SPEED, UNIQUE							
DEFAULT - NOERASE, NOREPL, NOREUSE, NOSPAN, NOWCK, SHR(1,3), UNORDER							
CATALOG - ICF.MJB.UCAT							
ASSOC - LABJ.MJB.TESTKD4.DATA (DATA)							
LABJ.MJB.TESTKD4.INDEX (INDEX)							
LABJ.MJB.TKD41.AIX (AIX)							
LABJ.MJB.TESTKD5	KSDS	LABS56	60	53			
DATA - RECSZ=215 CISZ=16384 FREESPACE%-CI=10 FREESPACE%-CA=10							
INDEX - KEYL=8 CISZ=512							
EXTENT -	1. LABS56	0056--00	0056--1D	TRKS=30			
EXTENT -	2. LABS56	0084--00	0084--1D	TRKS=30	INDEX		
ALLOC - DATA=(CYL,(1,1)) INDEX=(CYL,(1,1))							
STATS - EXCP=37 GET=400 TOT=200							
STATS - LAST MOD=NOV 19, 1987 1:50 PM LAST USED=NOV 15, 1987							
ATTR - IMBED, SPEED, UNIQUE							
DEFAULT - NOERASE, NOREPL, NOREUSE, NOSPAN, NOWCK, SHR(1,3), UNORDER							
CATALOG - ICF.MJB.UCAT							
ASSOC - LABJ.MJB.TESTKD5.DATA (DATA)							
LABJ.MJB.TESTKD5.INDEX (INDEX)							
LABJ.MJB.TKD51.AIX (AIX)							

***** CLUSTER TRACK ALLOCATION SUMMARY *****							
KSDS =	2	TRACKS =	120	USED =	106	IDLE =	14
TOTAL =	2	TRACKS =	120	USED =	106	IDLE =	14

Figure 4-8. Sample VSAM CLDATA Report

DISTRS

1996.228		AUG 15, 1996		VOLUME DISTRIBUTION SUMMARY										PAGE		
THURSDAY		1.10 PM												SAMS:DISK 9.1.0		
				FREE	--DISTRIBUTION BY SIZE OF DATA SET--					-----TRACKS-----				-----FREE SPACE-----		
VOLUME	TYPE	ADDR	STAT	DSCBS	ZERO	1-50	51-125	126-300	301->	DEVICE	ALLOC	AVAIL	IDLE	EXTENTS	THREE	LARGEST
-----				-----	-----					-----				-----		
WRK000	8598	0340	NONE	764		72	8	2	1	26550	2250	24300	473	18	10380	5565 348
-----				-----	-----					-----				-----		
TOTAL VOLS		1		764		72	8	2	1	26550	2250	24300	473	18		

Figure 4-9. Sample Volume Distribution DISTRs Report

DSINDEX

1987.329 NOV 25, 1987		D A T A S E T I N D E X		PAGE 1	
WEDNESDAY 9.32 AM				SAMS:Disk 8.2	
DATA SET NAME	VOLUME	DATA SET NAME	VOLUME	DATA SET NAME	VOLUME

LABS.DMS99.ASMLOG	LABS83	LABS.DMS99.USERREQS	LABS51		
LABS.DMS99.ASSEMSRC	LABS81	LABS.DMS99.WORKSHOP.CNTL	LABS83		
LABS.DMS99.CATNAMES	LABS51	LABS.DMS99.WORKSHOP.MANUAL.	LABS83		
LABS.DMS99.CHANGES	LABS55	ADSDOCS			
LABS.DMS99.COMMNET	LABS56	LABS.DMS99.WORKSHOP.MANUAL.	LABS83		
LABS.DMS99.COMMNET.LOADLIB	LABS83	OVERLAY			
LABS.DMS99.COMM12.LOADLIB	LABS56	LABS.DMS99.WORKSHOP.REFER	LABS55		
LABS.DMS99.DOCLIB	LABS50				
LABS.DMS99.DOCLIBID	LABS83				
LABS.DMS99.DOCUPD	LABS55				
LABS.DMS99.FIXLETR	LABS82				
LABS.DMS99.FOREIGN	LABS82				
LABS.DMS99.FX3AOUT	LABS54				
LABS.DMS99.FX3A1OUT	LABS54				
LABS.DMS99.FX3BOUT	LABS54				
LABS.DMS99.FX3COUT	LABS54				
LABS.DMS99.INSTALOG	LABS83				
LABS.DMS99.ISPSLIB	LABS55				
LABS.DMS99.LOADLIB	LABS83				
LABS.DMS99.OLDASPMR	LABS56				
LABS.DMS99.PARMLIB	LABS55				
LABS.DMS99.PMRSUMRY	LABS55				
LABS.DMS99.PROJECTS	LABS56				
LABS.DMS99.SLSS	LABS81				
LABS.DMS99.SOURCE	LABS83				
LABS.DMS99.SPFPANEL	LABS82				
LABS.DMS99.ST\$DUMP1	LABS56				
LABS.DMS99.ST\$SNAP	LABS56				
LABS.DMS99.STAFFRES	LABS51				
LABS.DMS99.SUPPTDOC	LABS82				
LABS.DMS99.TABLES	LABS56				
LABS.DMS99.TEL	LABS55				
LABS.DMS99.TELEX.AUS	LABS83				
LABS.DMS99.TELEX.CLOSED	LABS81				
LABS.DMS99.TELEX.CLOSEDX	LABS83				
LABS.DMS99.TELEX.ENG	LABS83				
LABS.DMS99.TELEX.ETC	LABS83				
LABS.DMS99.TELEX.ETCOLD	LABS83				
LABS.DMS99.TELEX.FRA	LABS83				
LABS.DMS99.TELEX.GER	LABS83				
LABS.DMS99.TELEX.GMBH	LABS82				
LABS.DMS99.TELEX.HKG	LABS83				
LABS.DMS99.TELEX.ISR	LABS83				
LABS.DMS99.TELEX.ITALY	LABS83				
LABS.DMS99.TELEX.JAPAN	LABS83				
LABS.DMS99.TELEX.MAL	LABS83				
LABS.DMS99.TELEX.SCAND	LABS82				
LABS.DMS99.TELEX.SCAND	LABS83				
LABS.DMS99.TELEX.SNG	LABS83				
LABS.DMS99.TELEX.STHAMER	LABS83				
LABS.DMS99.USERMODS	LABS83				
LABS.DMS99.USERMODS.BEFORE.	LABS82				
D27JAN87					

TOTAL DATA SETS		57			

Figure 4-10. Sample Data Set Index DSINDEX Report

DSUTIL

1990.100	APR 10, 1990	DATA SET UTILIZATION											PAGE	1		
TUESDAY	11.39 AM												SAMS:Disk	8.2		
FLAG	-----DATES-----					C P			A R OPEN		-----TRACKS-----					
DAYS	USED	DATE	MOD	DATE	CRE	DATE	EXP	DATE	DATA SET NAME	VOLUME	T O	COUNT	JOBNAME	ALLOC	USED	IDLE
1990.033	1990.033	1990.033							ISPKSV1.DISPOSE.A	ISP805	Y F		ISPKSV1	1		1
1990.033	1990.033	1990.033	1990.123						ISPKSV1.DISPOSE.B	ISP805	Y F		ISPKSV1	1		1
1990.070	1990.070	1990.010							ISPKSV1.TEST	DMSK01	Y			24	1	23
1990.053	1990.053	1990.053							ISPKSV1.TEST.ARCHIVE	DMSK02	N		ISPKSV1	1	1	
1990.079	1990.079	1990.079							ISPKSV1.TEST.COPY	DMSK02	Y F		ISPKSV1	3	1	2
1990.082	1990.082	1990.082							ISPKSV1.TEST.COPYA	DMSK01	Y F		ISPKSV1	3	1	2
1990.082	1990.082	1990.082							ISPKSV1.TEST.COPYA	DMSK02	- F		ISPKSV1	3	1	2
1990.082	1990.082	1990.082							ISPKSV1.TEST.COPYB	DMSK01	Y F		ISPKSV1	3	1	2
1990.055	1990.055	1990.055							ISPKSV1.TEST.DSUTIL	DMSK02	Y		ISPKSV1	1	1	
1989.205	1989.205	1989.205							ISPKSV1.TEST.REPRO	DMSE01	-		ISPKSV1	2	1	1
1990.088	1990.088	1990.088							LABIVP1.DLG.GDGMODEL.G0001V00	SMS501	Y F		ISPDLG1	1		1
1990.088	1990.088	1990.088							LABIVP1.DLG.GDGMODEL.G0003V00	SMS501	Y F		ISPDLG1	1		1
1990.089	1990.089	1990.089							LABIVP1.DLG.MVOL.PS2	SMS501	Y F		ISPDLG1M	18	18	
1990.039	1990.039	1990.039	1990.025						LABIVP1.DUMMYA	SMS501	Y F			1		1
1990.100	1990.100	1990.072							LABIVP1.EDP.COMPRES1	SMS501	Y F		ISPEDP1A	3	1	2
1990.089	1990.089	1990.089							LABIVP1.GLG.TESTNONE	SMS501	Y F		ISPGLG1	1	1	
1990.089	1990.089	1990.089							LABIVP1.GLG.TEST1	SMS501	Y F		ISPGLG1	3	1	2
1990.024	1990.024	1990.024							LABIVP1.IDLE.A	SMS501	Y F		ISPKSV1	25	1	24
1990.088	1990.088	1990.088							LABIVP1.JULDUMMY	SMS501	Y F		ISPJUL1	5	1	4
1990.088	1990.088	1990.088							LABIVP1.JUL1	SMS501	Y F		ISPJUL1	3		3
1990.089	1990.089	1990.089							LABIVP1.KSV.DEFAULT	SMS501	Y F		ISPKSV1	3		3
1990.033	1990.033	1990.033	1990.123						LABIVP1.KSV.PDS	SMS501	Y		ANYTHING	28	1	27
1990.003	1990.003	1990.003							LABIVP1.KSV.PRIM.DMSC.	SMS501	- F			1	1	
									D1990003.T103922							
1990.003	1990.003	1990.003							LABIVP1.KSV.PRIM.DMSC.	SMS501	N F			1	1	
									D1990003.T110327							
1990.003	1990.003	1990.003							LABIVP1.KSV.PRIM.DMSC.	SMS501	N F			1	1	
									D1990003.T110641							
1989.347	1989.347	1989.347							LABIVP1.KSV.SEQGDG02.G0001V00	SMS501	Y F			25	1	24
1989.347	1989.347	1989.347							LABIVP1.KSV.SEQGDG02.G0003V00	SMS501	Y F			25	1	24
1989.347	1989.347	1989.347							LABIVP1.KSV.SEQTEST	SMS501	- F		ISPKSV1	25	1	24
1990.002	1990.002	1990.002							LABIVP1.KSV.SEQTESTB	SMS501	- F		ISPKSV1	3		3
1990.002	1990.002	1990.002							LABIVP1.KSV.SEQTESTC	SMS501	- F		ISPKSV1	3		3
1990.032	1990.032	1990.032							LABIVP1.ONCE	SMS501	Y F		TSSIMP	3		3
1990.032	1990.032	1990.032							LABIVP1.SIMP	SMS501	Y		TSSIMP	3		3
1990.085	1990.085	1990.085							LABIVP1.SRB.FILES.B.D1990085.	SMS501	Y F			1	1	
									T150300							
1990.086	1990.086	1990.086							LABIVP1.SRB.FILES.B.D1990086.	SMS501	Y F			1	1	
									T081300							
TOTAL DATA SETS										34	TOTAL TRACKS ALLOC		226			
* - DATA SET HAS NOT BEEN USED WITHIN VALUE SPECIFIED IN FLAGDAYS PARAMETER.											USED			40		
\$ - LAST USE DATE IS ZERO; VALUE SHOWN IS CREATION DATE.											IDLE				186	

Figure 4-11. Sample Data Set Utilization DSUTIL Report

FREESP

1996.228	AUG 15, 1996	VOLUME FREE SPACE DETAIL										PAGE			
THURSDAY	1.10 PM											SAMS:DISK 9.1.0			
VOLUME WRK000		DEVICE 8598		ADDR 0340		THIS VOLUME IS NOT SMS MANAGED									
TRACK	LEN	TRACK	LEN	TRACK	LEN	TRACK	LEN	TRACK	LEN	TRACK	LEN	TRACK	LEN	TRACK	LE

22	1														
149	1														
241	17														
658	1														
702	3														
735	15														
765	239														
1005	1281														
3090	1457														
4641	24														
4666	10380														
15061	14														
15090	5565														
20795	85														
21010	55														
21133	2														
21315	1680														
23070	3480														

TOTAL FREE TRACKS				24300											

Figure 4-12. Sample Volume Free Space FREESP Report

LAYOUT

1996.228		AUG 15, 1996		V O L U M E L A Y O U T				PAGE	
THURSDAY		1.10 PM						SAMS:DISK 9.1.0	
VOLUME WRK000		DEVICE 8598		ADDR 0340		THIS VOLUME IS NOT SMS MANAGED			
D - DATA EXTENT		S - SHARED CYLINDER		F - FREE SPACE		V - VTOC			
TYP	LEN	TYP	LEN	TYP	LEN	TYP	LEN	TYP	LEN
-----		-----		-----		-----		-----	
V	17	D	15	D	1				
D	3	D	20	D	1				
F	1	D	2	D	1				
D	1	D	2	D	1				
D	21	D	3	D	15				
D	8	F	3	D	150				
D	10	D	15	F	1680				
D	1	D	15	D	75				
D	1	F	15	F	3480				
D	1	D	15						
D	4	F	239						
D	1	D	1						
D	4	F	1281						
D	30	D	804						
D	30	F	1457						
D	3	D	1						
D	1	D	1						
D	5	D	1						
D	1	D	1						
D	3	D	1						
D	1	D	8						
F	1	D	75						
D	60	D	6						
D	1	F	24						
D	1	D	1						
F	17	F	10380						
D	14	D	15						
D	61	F	14						
D	4	D	15						
D	249	F	5565						
D	6	D	60						
D	1	F	85						
D	5	D	75						
D	1	D	30						
D	1	D	25						
D	1	F	55						
D	1	D	58						
D	1	D	5						
D	6	D	5						
D	2	F	2						
D	2	D	1						
D	15	D	3						
D	1	D	1						
D	1	D	1						
D	1	D	1						
D	5	D	1						
D	5	D	1						
F	1	D	1						
D	1	D	1						

Figure 4-13. Sample Volume Layout Report

MAPPER

1996.228	AUG 15, 1996	VOLUME MAP										PAGE		
THURSDAY	1.10 PM											SAMS:DISK 9.1.0		
VOLUME WRK000	DEVICE 8598	ADDR 0340	THIS VOLUME IS NOT SMS MANAGED											
DATA SET NAME		CC	HH	TRACK	LEN	EXT	DATA SET NAME		CC	HH	TRACK	LEN	EXT	

VTOC		0000	02	2	17	1	STRL001.D27		002E	05	695	2	1	
STRL002.DECK		0001	04	19	3	1	STRL001.D29		002E	07	697	2	1	
FREE SPACE		0001	07	22	1		STRL001.D35		002E	09	699	3	1	
STRL000.RERTEST.JCL		0001	08	23	1	1	**FREE SPACE**		002E	0C	702	3		
STRL.DISKV788.INSTALL		0001	09	24	21	1	STRL000.TESTKSDS.ALTINDX.CLUSTER.D@		002F	00	705	15	1	
STRL.DISKV788.PROCLIB		0003	00	45	8	1	STRL005.TEST.PS		0030	00	720	15	1	
STRL001.PETOUT		0003	08	53	10	1	**FREE SPACE**		0031	00	735	15		
STRL005.T91.EMPTYTPS		0004	03	63	1	1	STRL000.FILES.UNLOAD		0032	00	750	15	1	
DISK910.WORK.ARESPARM		0004	04	64	1	1	**FREE SPACE**		0033	00	765	239		
STRL005.T91.EXP.KS07011.I@		0004	05	65	1	1	STRL005.SRS.VK.DATA		0042	0E	1004	1	1	
STRL001.D21		0004	06	66	4	1	**FREE SPACE**		0043	00	1005	1281		
STRL001.TSOTEMP2		0004	0A	70	1	1	STRL.DISKV788.SOURCE		0098	06	2286	804	1	
STRL005.T91.COMPRESS.PDS		0004	0B	71	4	1	**FREE SPACE**		00CE	00	3090	1457		
STRL005.T91.PDS		0005	00	75	30	1	STRL005.SRS.VK21.DATA		012F	02	4547	1	1	
STRL005.T91.PDS.COPY2		0007	00	105	30	1	STRL005.SRS.VK21.INDEX		012F	03	4548	1	1	
STRL001.D06		0009	00	135	3	1	STRL002.WORK37.WORKFILE		012F	04	4549	1	1	
STRL000.TESTKSDS.BASE.CLUSTER.I@		0009	03	138	1	1	STRL001.D01		012F	05	4550	1	6	
STRL002.WORK05.WORFILE		0009	04	139	5	1	STRL001.D01		012F	06	4551	1	7	
STRL002.WORK12.WORFILE		0009	09	144	1	1	STRL001.D03		012F	07	4552	8	1	
STRL002.TEST.CMPRS15		0009	0A	145	3	1	STRL005.T91.JUNK1.KSDS.D@		0130	00	4560	75	1	
STRL002.WORK43.WORKFILE		0009	0D	148	1	1	STRL005.B91.GDGTST.G0002V00		0135	00	4635	6	1	
FREE SPACE		0009	0E	149	1		**FREE SPACE**		0135	06	4641	24		
STRL003.COMPWORK.WORK99		000A	00	150	60	1	STRL005.SRS.VK.INDEX		0137	00	4665	1	1	
STRL002.DISKARCH.DMSC.D1996227.T181725		000E	00	210	1	1	**FREE SPACE**		0137	01	4666	10380		
STRL002.DISKARCH.DMSC.D1996227.T182956		000E	01	211	1	1	STRL005.PDS.TEST		03EB	01	15046	15	1	
FREE SPACE		0010	01	241	17		**FREE SPACE**		03EC	01	15061	14		
STRL.DISKV788.PARMLIB		0011	03	258	14	1	STRL005.T91.OUTPUT.FILES		03ED	00	15075	15	1	
STRL.DISKV788.MACLIB		0012	02	272	61	1	**FREE SPACE**		03EE	00	15090	5565		
STRL.DISKV788.DOCLIB		0016	03	333	4	1	STRL000.TESTKSDS.BASE.CLUSTER.D@		0561	00	20655	60	1	
STRL.DISKV788.LOADXXX		0016	07	337	249	1	**FREE SPACE**		056A	05	20795	85		
STRL001.D05		0027	03	588	6	1	STRL005.T91.JUNK1.ESDS.D@		0570	00	20880	75	1	
STRL000.TESTKSDS.ALTINDX.CLUSTER.I@		0027	09	594	1	1	STRL005.T91.PS		0575	00	20955	30	1	
STRL005.VK.INDEX		0027	0A	595	5	1	STRL001.D34		0577	00	20985	25	1	
STRL005.T91.EMPTYTPDS		0028	00	600	1	1	**FREE SPACE**		0578	0A	21010	55		
STRL000.TEST.DATASET		0028	01	601	1	1	STRL000.FTP.DECK		057C	05	21065	58	1	
STRL005.T91.EXP.PO99365		0028	02	602	1	1	STRL001.PETOUT		0580	03	21123	5	4	
STRL005.T91.EXP.KS00000.D@		0028	03	603	1	1	STRL001.PETOUT		0580	08	21128	5	5	
STRL005.T91.EXP.KS00000.I@		0028	04	604	1	1	**FREE SPACE**		0580	0D	21133	2		
STRL005.B91.GDGTST.G0001V00		0028	05	605	6	1	STRL001.D01		0581	00	21135	1	1	
STRL003.TEST.PO1		0028	0B	611	2	1	STRL000.RELEASE.TEST		0581	01	21136	3	1	
STRL003.TEST.PO4		0028	0D	613	2	1	STRL001.D01		0581	04	21139	1	2	
STRL005.VK.DATA		002A	00	630	15	1	STRL001.D01		0581	05	21140	1	3	
STRL005.T91.JUNK1.KSDS.I@		002B	00	645	1	1	STRL001.D01		0581	06	21141	1	4	
STRL002.WORK33.WORKFILE		002B	01	646	1	1	STRL001.D01		0581	07	21142	1	5	
STRL005.T91.EXP.PO00000		002B	02	647	1	1	STRL005.T91.EXP.KS97001.D@		0581	08	21143	1	1	
STRL001.PETOUT		002B	03	648	5	2	STRL005.T91.EXP.KS97001.I@		0581	09	21144	1	1	
STRL001.PETOUT		002B	08	653	5	3	STRL005.T91.EXP.KS98002.D@		0581	0A	21145	1	1	
FREE SPACE		002B	0D	658	1		STRL005.T91.EXP.KS98002.I@		0581	0B	21146	1	1	
STRL000.D		002B	0E	659	1	1	STRL005.T91.EXP.KS99365.D@		0581	0C	21147	1	1	
STRL000.FILES910.REGRESS.UNLOAD		002C	00	660	15	1	STRL005.T91.EXP.KS99365.I@		0581	0D	21148	1	1	
STRL001.D1D		002D	00	675	20	1	STRL005.T91.EXP.KS07011.D@		0581	0E	21149	1	1	

Figure 4-14. Sample Volume Map Mapper Report

MEMBER

1987.329 NOV 25, 1987		P A R T I T I O N E D D A T A S E T D I R E C T O R Y						PAGE	1
WEDNESDAY 9.32 AM								SAMS:Disk	8.2
DATA SET NAME	LABS.MJB.SOURCE				VOLUME LABS83	DEVICE 3380	ADDR 1A3		
MEMBER	MEMBER	MEMBER	MEMBER	MEMBER	MEMBER	MEMBER	MEMBER	MEMBER	MEMBER
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----
\$INIT	SPFCODE								
ADSDM263	SPLITREC								
ADSMIDEL	TESTPGM								
ADSUT410	TESTTIME								
ADSUT411	TRANSBAR								
ASM	UT190								
ASMSTD	VS032								
BILLING	WRLRGDA								
CLOPT	XDM175								
CONVERT	XMSNAP								
CSTLTR	ZCOLL								
CSTLTR2	ZDISK								
DATA									
DATECONV									
DM002									
D3380									
D3380FIX									
D3380K									
FICHE									
GBM1									
GMB									
HEXPRINT									
HEXPRT									
IIIDM192									
IIIDM260									
IIIDM263									
IIIDM265									
IIIDM466									
IIIUT110									
IIIVS032									
IIIVS088									
ISFPARMS									
ISFPGEN									
MAINT									
NOTES									
PRINTHI									
QUITCLM									
RECON1									
RECON2									
RESAFTEX									
RESPRIEX									
RETAFTEX									
RETPRIEX									
R75DM260									
R75DM263									
SCRTCHEX									
SHRINKCP									
SHRINKDC									
SHRKCP1									
SMFSELCT									
SMF1									

TOTAL MEMBERS				63					

Figure 4-15. Sample PDS MEMBER Report

MVDICT

1996.228	AUG 15, 1996	M U L T I P L E V O L U M E D A T A S E T D I C T I O N A R Y													PAGE						
THURSDAY	1.10 PM														SAMS:DISK 9.1.0						
															C P						
															DEVICE UNIT A R						
															---ALLOCATION---		---TRACKS---				
DATA SET NAME	VOLUME	TYPE	ADDR	T O	DSO	RECFM	BLKSZ	LRECL	TYP	PRIM	SEC	ALLOC	IDLE	EXT	CREDATE	EXPDATE	USEDATE				

STRL005.B91.GDGTST. G0002V00	WRK000	8598	0340	Y	PS	FB	19040	80	TRK	6		6	6	1	1996.214						
STRL005.PDS.TEST	WRK000	8598	0340	Y	F	PO	FB	9040	80	TRK	15	15	15	14	1	1996.218					
STRL005.SRS.VE.DATA	WRK000	8598	0340	Y	R	AM	U	4096		CYL	1	1	15		1	1996.218	1999.365				
STRL005.SRS.VK.DATA	WRK000	8598	0340	Y	R	AM	U	4096		TRK	1	1	1		1	1996.218	1999.365				
STRL005.SRS.VK.INDEX	WRK000	8598	0340	Y	R	AM	U	1024		TRK	1	1	1		1	1996.218	1999.365				
STRL005.SRS.VK21.DATA	WRK000	8598	0340	Y	R	AM	U	4096		TRK	1	1	1		1	1996.218	1999.365				
STRL005.SRS.VK21.INDEX	WRK000	8598	0340	Y	R	AM	U	1024		TRK	1	1	1		1	1996.218	1999.365				
STRL005.TEST.FILES	WRK000	8598	0340	Y	F	DA	F	23476		CYL	10		150		1	1996.218					
STRL005.TEST.PS	WRK000	8598	0340	Y	F	PS	FB	9040	80	TRK	15	15	15	11	1	1996.218					
STRL005.T91.COMPRESS.PDS	WRK000	8598	0340	Y	F	PO	FB	6240	80	TRK	4	1	4		1	1996.221					
STRL005.T91.EMPTYTYPDS	WRK000	8598	0340	Y	F	PO	FB	800	80	TRK	1	1	1		1	1996.227					
STRL005.T91.EMPTYTYP	WRK000	8598	0340	Y	F	PS	FB	800	80	TRK	1	1	1	1		1996.227	1996.227				
STRL005.T91.EXP.KS00000.D@	WRK000	8598	0340	Y	R	AM	U	2048		TRK	1	1	1		1	1996.220	1999.365				
STRL005.T91.EXP.KS00000.I@	WRK000	8598	0340	Y	R	AM	U	512		TRK	1	1	1		1	1996.220	1999.365				
STRL005.T91.EXP.KS07011.D@	WRK000	8598	0340	Y	R	AM	U	2048		TRK	1	1	1		1	1996.220	1999.365				
STRL005.T91.EXP.KS07011.I@	WRK000	8598	0340	Y	R	AM	U	512		TRK	1	1	1		1	1996.220	1999.365				
STRL005.T91.EXP.KS97001.D@	WRK000	8598	0340	Y	R	AM	U	2048		TRK	1	1	1		1	1996.220	1999.365				
STRL005.T91.EXP.KS97001.I@	WRK000	8598	0340	Y	R	AM	U	512		TRK	1	1	1		1	1996.220	1999.365				
STRL005.T91.EXP.KS98002.D@	WRK000	8598	0340	Y	R	AM	U	2048		TRK	1	1	1		1	1996.220	1999.365				
STRL005.T91.EXP.KS98002.I@	WRK000	8598	0340	Y	R	AM	U	512		TRK	1	1	1		1	1996.220	1999.365				
STRL005.T91.EXP.KS99365.D@	WRK000	8598	0340	Y	R	AM	U	2048		TRK	1	1	1		1	1996.220	1999.365				
STRL005.T91.EXP.KS99365.I@	WRK000	8598	0340	Y	R	AM	U	512		TRK	1	1	1		1	1996.220	1999.365				
STRL005.T91.EXP.PO00000	WRK000	8598	0340	Y	F	PO	FB	3120	80	TRK	1	1	1		1	1996.220					
STRL005.T91.EXP.PO99365	WRK000	8598	0340	Y	F	PO	FB	3120	80	TRK	1	1	1		1	1996.220	1999.365				
STRL005.T91.JUNK1.ESDS.D@	WRK000	8598	0340	Y	R	AM	U	4096		CYL	5	1	75		1	1996.219	1999.365				
STRL005.T91.JUNK1.KSDS.D@	WRK000	8598	0340	Y	R	AM	U	4096		CYL	5	1	75		1	1996.219	1999.365				
STRL005.T91.JUNK1.KSDS.I@	WRK000	8598	0340	Y	R	AM	U	2048		TRK	1	1	1		1	1996.219	1999.365				
STRL005.T91.JUNK1.RRDS.D@	WRK000	8598	0340	Y	R	AM	U	4096		CYL	5	1	75		1	1996.219	1999.365				
STRL005.T91.OUTPUT.FILES	WRK000	8598	0340	Y	F	DA	F	23476		CYL	1		15		1	1996.220					
STRL005.T91.PDS	WRK000	8598	0340	Y	F	PO	FB	9040	80	CYL	2	1	30	25	1	1996.219					
STRL005.T91.PDS.COPY2	WRK000	8598	0340	Y	F	PO	FB	9040	80	CYL	2	1	30	25	1	1996.219					
STRL005.T91.PS	WRK000	8598	0340	Y	F	PS	FB	9040	80	CYL	2	1	30	26	1	1996.219					
STRL005.VK.DATA	WRK000	8598	0340	Y	R	AM	U	4096		CYL	1	1	15		1	1996.221	1999.365				
STRL005.VK.INDEX	WRK000	8598	0340	Y	R	AM	U	2048		TRK	5	1	5		1	1996.221	1999.365				

TOTAL DATA SETS		78											TOTAL ALLOCATED TRACKS			2105					
															TOTAL IDLE TRACKS			349			

Figure 4-16. Sample MVDICT Report

PDSTAT

1996.228 AUG 15, 1996		P A R T I T I O N E D D A T A S E T S T A T U S										PAGE					
THURSDAY 1.10 PM												SAMS:DISK 9.1.0					
DATA SET NAME	VOLUME	DEVICE --SPACE REQUEST--				-----TRACKS-----				EXT	-DIRECTORY BLOCKS-			--DIRECTORY ENTRIES--			
		TYPE	TYPE	PRIM	SEC	ALLOC	USED	IDLE	ALLOC		USED	IDLE	ALLOC	USED	IDLE	LE	
STRL.DISKV788.DOC LIB	WRK000	8598	TRK	4	25	4	4		1	164	1	163					
STRL.DISKV788.INSTALL	WRK000	8598	TRK	21	10	21	21		1	65	10	55		714	102	612	2
STRL.DISKV788.LOADXXX	WRK000	8598	TRK	249	250	249	249		1	250	200	50		1749	1393	356	3
STRL.DISKV788.MAC LIB	WRK000	8598	TRK	61	25	61	61		1	164	28	136		2118	343	1775	1
STRL.DISKV788.PARMLIB	WRK000	8598	TRK	14	5	14	9	5	1	12	4	8		228	60	168	1
STRL.DISKV788.PROCLIB	WRK000	8598	TRK	8	5	8	8		1	31	3	28		431	40	391	1
STRL.DISKV788.SOURCE	WRK000	8598	TRK	804	25	804	804		1	250	69	181		4807	1352	3455	1
STRL000.RERTEST.JCL	WRK000	8598	BLK	12	2	1	1		1	1	1			5	1	4	4
STRL002.TEST.CMPRS15	WRK000	8598	BLK	36	30	3	1	2	1	25	1	24					
STRL003.TEST.PO1	WRK000	8598	TRK	2	2	2	1	1	1	5	1	4		104	9	95	1
STRL003.TEST.PO4	WRK000	8598	TRK	2	1	2	1	1	1	5	1	4		104	9	95	1
STRL005.PDS.TEST	WRK000	8598	TRK	15	15	15	1	14	1	2	1	1					
STRL005.T91.COMPRESS.PDS	WRK000	8598	TRK	4	1	4	4		1	7	7			42	41	1	4
STRL005.T91.EMPTY PDS	WRK000	8598	TRK	1	1	1	1		1	10	1	9					
STRL005.T91.EXP.PO00000	WRK000	8598	TRK	1	1	1	1		1	1	1			5	1	4	4
STRL005.T91.EXP.PO99365	WRK000	8598	TRK	1	1	1	1		1	1	1			5	1	4	4
STRL005.T91.PDS	WRK000	8598	CYL	2	1	30	5	25	1	10	1	9		209	1	208	1
STRL005.T91.PDS.COPY2	WRK000	8598	CYL	2	1	30	5	25	1	10	1	9		209	1	208	1

TOTAL DATA SETS		18		TOTAL TRACKS		ALLOC		1251									
										USED		1178					
										IDLE		73					

Figure 4-17. Sample PDSTAT Report

POCCHH

1996.228	AUG 15, 1996	PARTITIONED DATA SET DIRECTORY (CCHHR SEQUENCE)										PAGE
THURSDAY	1.10 PM											SAMS:DISK 9.1.0
DATA SET NAME		STRL.DISKV788.INSTALL				VOLUME WRK000		DEVICE 8598				
C C H H R	MEMBER	LTH	T T R C	U S E R		D A T A						

0001000D1C	DRESTORE	12	00041C00									
0001000D1F	SARIPL	12	00041F00									
0001000E01	DUMPTRK	12	00050100									
0001000E03	RMAP	12	00050300									
0001000E05	RELKMI2	12	00050500									
0001000E07	RELKSHRK	12	00050700									
0001000E09	T@TARCE	42	0005090F	00910117	FE400091	01171634	2499000A	000A0000	000000E3	D6D2E8D6	F440	
0001000E0B	T@ARCHIV	42	00050B0F	03910129	FE400090	11261129	15430051	00510006	000000E3	D6D2E8D6	F440	
0001000E0E	T@ARCRES	42	00050E0F	02910129	FE400090	11261129	57940099	00990003	000000E3	D6D2E8D6	F440	
0001000E12	T@DMSRPT	42	0005120F	02910129	FE400091	01111130	2992004E	004F0003	000000E3	D6D2E8D6	F440	
0001000E15	T@DSCBUP	42	0005150F	0C910129	FE400090	0091252F	1832002F	00A50033	E2C2C5C4	D74040D6	F440	

TOTAL MEMBERS REPORTED ON				11								

Figure 4-18. Sample POCCHH Report

PODUMP

1987.329	NOV 25, 1987	PARTITIONED DATA SET DIRECTORY (MEMBER NAME SEQUENCE)										PAGE
WEDNESDAY	9.32 AM											1
DATA SET NAME		LABS.MJB.SOURCE		VOLUME LABS83		DEVICE 3380						
MEMBER	LTH	C C H H R	T T R C	U S E R		D A T A						
\$INIT	42	02A1000C0F	00000F0F	01050000	0087231F	0087231F	151100CB	00CB0000	E2C2D4D1	C2404040	4040	
ADSDM263	42	02A5000502	0035020F	01000000	0087328F	0087328F	12300840	08400000	E2C2D4D1	C2404040	4040	
ADSMIDEL	42	02A1000C14	0000140F	01020000	0085144F	0085144F	08510497	0097044C	E2C2D4D1	C2404040	4040	
ADSUT410	42	02A2000001	0003010F	01010000	0087244F	0087244F	151000B0	00B00002	E2C2D4D1	C2404040	4040	
ADSUT411	42	02A2000006	0003060F	01000000	0087244F	0087244F	15170088	00880000	E2C2D4D1	C2404040	4040	
ASM	42	02A200000A	00030A0F	01020000	0085170F	0086056F	15020009	00080000	E2C2D4D1	C2404040	4040	
SPLITREC	42	02A3000014	0012140F	01010000	0085270F	0085270F	15300024	00240003	E2C2D4D1	C2404040	4040	
TESTPGM	42	02A3000102	0013020F	01070000	0086065F	0086211F	1646003B	00290024	E2C2D4D1	C2404040	4040	
TESTTIME	42	02A3000105	0013050F	01000000	0087189F	0087189F	1335001C	001C0000	E2C2D4D1	C2404040	4040	
TRANSBAR	42	02A3000107	0013070F	01000000	0087054F	0087054F	1505003B	003B0000	E2C2D4D1	C2404040	4040	
WRLRGDA	42	02A3000303	0015030F	01000000	0087288F	0087288F	09070044	00440000	E2C2D4D1	C2404040	4040	
XDML75	42	02A3000306	0015060F	01000000	0086289F	0086289F	19510002	00020000	E2C2D4D1	C2404040	4040	
TOTAL MEMBERS REPORTED ON				12								

Figure 4-19. Sample PODUMP Report

POTTRD

1996.228	AUG 15, 1996		PARTITIONED DATA SET DIRECTORY										(TTR SEQUENCE)		PAGE		
THURSDAY	1.10 PM																SAMS:DISK 9.1.0
DATA SET NAME	STRL.DISKV788.INSTALL		VOLUME WRK000										DEVICE 8598				
T T R C	MEMBER	LTH	C C H H R	U S E R												D A T A	

00041C00	DRESTORE	12	0001000D1C														
00041F00	SARIPL	12	0001000D1F														
00050100	DUMPTRK	12	0001000E01														
00050300	RMAP	12	0001000E03														
00050500	RELKMI2	12	0001000E05														
00050700	RELKSHRK	12	0001000E07														
0005090F	T@TARCE	42	0001000E09	00910117	FE400091	01171634	2499000A	000A0000	000000E3	D6D2E8D6	F440						
00050B0F	T@ARCHIV	42	0001000E0B	03910129	FE400090	11261129	15430051	00510006	000000E3	D6D2E8D6	F440						
00050E0F	T@ARCRES	42	0001000E0E	02910129	FE400090	11261129	57940099	00990003	000000E3	D6D2E8D6	F440						
0005120F	T@DMSRPT	42	0001000E12	02910129	FE400091	01111130	2992004E	004F0003	000000E3	D6D2E8D6	F440						
0005150F	T@DSCBUP	42	0001000E15	0C910129	FE400090	0091252F	1832002F	00A50033	E2C2C5C4	D74040D6	F440						

TOTAL MEMBERS REPORTED ON				11													

Figure 4-20. Sample POTTRD Report

POXREF

1987.329	NOV 25, 1987	DATA SET NAME CROSS REFERENCE FOR PDS MEMBER NAMES				PAGE	1
WEDNESDAY	9.32 AM					SAMS:Disk	8.2
MEMBER	DATA SET NAME	VOLUME	DATA SET NAME	VOLUME			

\$INSTALL	LABS.DMS99.INSTALL	MVS002	SYS2.DMS.INSTALL	LABS56			
ADSAR001	LABS.DMS99.INSTALL	MVS002	SYS2.DMS.INSTALL	LABS56			
ADSAR002	LABS.DMS99.INSTALL	MVS002	SYS2.DMS.INSTALL	LABS56			
ADSAR003	LABS.DMS99.INSTALL	MVS002	SYS2.DMS.INSTALL	LABS56			
ADSAR004	LABS.DMS99.INSTALL	MVS002	SYS2.DMS.INSTALL	LABS56			
ADSAR005	LABS.DMS99.INSTALL	MVS002	SYS2.DMS.INSTALL	LABS56			
ADSAR007	LABS.DMS99.INSTALL	MVS002	SYS2.DMS.INSTALL	LABS56			
ADSAR008	LABS.DMS99.INSTALL	MVS002	SYS2.DMS.INSTALL	LABS56			
ADSAR010	LABS.DMS99.INSTALL	MVS002	SYS2.DMS.INSTALL	LABS56			
ADSMVS60	LABS.DMS99.INSTALL	MVS002	SYS2.DMS.INSTALL	LABS56			
ADSTH012	LABS.DMS99.INSTALL	MVS002	SYS2.DMS.INSTALL	LABS56			

TOTAL MEMBERS REPORTED ON		11					

Figure 4-21. Sample POXREF Report

PO1REF

1987.342	DEC 08, 1987	PDS CROSS REFERENCE FOR MEMBERS IN ONLY ONE DATA SET				PAGE	1
TUESDAY	9.39 AM					SAMS:Disk	8.2
MEMBER	DATA SET NAME	VOLUME	MEMBER	DATA SET NAME	VOLUME		
-----	-----	-----	-----	-----	-----		
\$\$\$\$\$DMS	LABS.DMS99.LOADLIB	LABS83	ADSMI003	LABS.DMS99.LOADLIB	LABS83		
ADSMI006	LABS.DMS99.LOADLIB	LABS83	ADSMI008	LABS.DMS99.LOADLIB	LABS83		
ADSMI010	LABS.DMS99.LOADLIB	LABS83	ADSMI011	LABS.DMS99.LOADLIB	LABS83		
ADSMI013	LABS.DMS99.LOADLIB	LABS83	ADSMI014	LABS.DMS99.LOADLIB	LABS83		
ADSMI015	LABS.DMS99.LOADLIB	LABS83	ADSMI016	LABS.DMS99.LOADLIB	LABS83		
ADSMI017	LABS.DMS99.LOADLIB	LABS83	ADSMI018	LABS.DMS99.LOADLIB	LABS83		
ADSMI019	LABS.DMS99.LOADLIB	LABS83	ADSMI025	LABS.DMS99.LOADLIB	LABS83		
ADSMI026	LABS.DMS99.LOADLIB	LABS83	ADSMI029	LABS.DMS99.LOADLIB	LABS83		
ADSMI031	LABS.DMS99.LOADLIB	LABS83	ADSMI033	LABS.DMS99.LOADLIB	LABS83		
ADSMI034	LABS.DMS99.LOADLIB	LABS83	ADSMI035	LABS.DMS99.LOADLIB	LABS83		
ADSMI036	LABS.DMS99.LOADLIB	LABS83	ADSMI037	LABS.DMS99.LOADLIB	LABS83		
ADSMI038	LABS.DMS99.LOADLIB	LABS83	ADSMI039	LABS.DMS99.LOADLIB	LABS83		
ADSMI040	LABS.DMS99.LOADLIB	LABS83	ADSMI041	LABS.DMS99.LOADLIB	LABS83		
ADSMI042	LABS.DMS99.LOADLIB	LABS83	ADSMI043	LABS.DMS99.LOADLIB	LABS83		
ADSMI044	LABS.DMS99.LOADLIB	LABS83	ADSMI045	LABS.DMS99.LOADLIB	LABS83		
ADSMI046	LABS.DMS99.LOADLIB	LABS83	ADSMI049	LABS.DMS99.LOADLIB	LABS83		
ADSMI054	LABS.DMS99.LOADLIB	LABS83	ADSMI055	LABS.DMS99.LOADLIB	LABS83		
ADSMI056	LABS.DMS99.LOADLIB	LABS83	ADSMI057	LABS.DMS99.LOADLIB	LABS83		
ADSMI059	LABS.DMS99.LOADLIB	LABS83	ADSMI060	LABS.DMS99.LOADLIB	LABS83		
ADSMI061	LABS.DMS99.LOADLIB	LABS83	ADSMI062	LABS.DMS99.LOADLIB	LABS83		
ADSMI063	LABS.DMS99.LOADLIB	LABS83	ADSMI101	LABS.DMS99.LOADLIB	LABS83		
ADSMI901	LABS.DMS99.LOADLIB	LABS83	ADSMI906	LABS.DMS99.LOADLIB	LABS83		
DMSPARM1	LABS.DMS99.LOADLIB	LABS83	DMSPARM2	LABS.DMS99.LOADLIB	LABS83		
JFF0024D	LABS.DMS99.SOURCE	LABS83					

TOTAL MEMBERS REPORTED ON		47					

Figure 4-22. Sample PO1REF Report

PO2REF

1987.329	NOV 25, 1987	PDS CROSS REFERENCE FOR MEMBERS IN TWO OR MORE DATA SETS			PAGE	1
WEDNESDAY	9.32 AM				SAMS:Disk	8.2
MEMBER	DATA SET NAME	VOLUME	DATA SET NAME	VOLUME		
-----	-----	-----	-----	-----		
ADSAR001	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
ADSAR002	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
ADSAR003	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
ADSAR004	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
ADSAR005	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
ADSAR007	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
ADSAR008	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
ADSAR010	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
ADSMVS60	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
ADSTH012	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
ADS0024D	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
ADS0124D	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
DARCHIVE	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
DERASE	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
DMSTABLE	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
DRESTORE	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
IDATMSTP	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
IFGOEX0A	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
JFF0024D	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
KBCOEX0A	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
LISTDMS	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
LISTREQ	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
NULLFILE	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		
RESTORE	LABS.DMS99.INSTALL	MVS002	LABS.DMS99.SOURCE	LABS83		

TOTAL MEMBERS REPORTED ON		24				

Figure 4-23. Sample PO2REF Report

SVDICT

S I N G L E V O L U M E D A T A S E T D I C T I O N A R Y												PAGE															
11996.228 AUG 15, 1996												SAMS:DISK 9.1.0															
THURSDAY 1.10 PM																											
C P																											
D E V I C E U N I T A R																											
---ALLOCATION--- ---TRACKS---																											
DATA SET NAME	VOLUME	TYPE	ADDR	T	O	DSO	RECFM	BLKSZ	LRECL	TYP	PRIM	SEC	ALLOC	IDLE	EXT	CREDATE	EXPDATE	USEDATE									
STRL005.B91.GDGTTEST. G0002V00	WRK000	8598	0340	Y	PS	FB		19040	80	TRK		6		6	6	1	1996.214										
STRL005.PDS.TEST	WRK000	8598	0340	Y	F	PO	FB	9040	80	TRK		15	15	15	14	1	1996.218										
STRL005.SRS.VE.DATA	WRK000	8598	0340	Y	R	AM	U	4096		CYL		1	1	15		1	1996.218	1999.365									
STRL005.SRS.VK.DATA	WRK000	8598	0340	Y	R	AM	U	4096		TRK		1	1	1		1	1996.218	1999.365									
STRL005.SRS.VK.INDEX	WRK000	8598	0340	Y	R	AM	U	1024		TRK		1	1	1		1	1996.218	1999.365									
STRL005.SRS.VK21.DATA	WRK000	8598	0340	Y	R	AM	U	4096		TRK		1	1	1		1	1996.218	1999.365									
STRL005.SRS.VK21.INDEX	WRK000	8598	0340	Y	R	AM	U	1024		TRK		1	1	1		1	1996.218	1999.365									
STRL005.TEST.FILES	WRK000	8598	0340	Y	F	DA	F	23476		CYL		10		150		1	1996.218										
STRL005.TEST.PS	WRK000	8598	0340	Y	F	PS	FB	9040	80	TRK		15	15	15	11	1	1996.218										
STRL005.T91.COMPRESS.PDS	WRK000	8598	0340	Y	F	PO	FB	6240	80	TRK		4	1	4		1	1996.221										
STRL005.T91.EMPTYPDS	WRK000	8598	0340	Y	F	PO	FB	800	80	TRK		1	1	1		1	1996.227										
STRL005.T91.EMPTYPS	WRK000	8598	0340	Y	F	PS	FB	800	80	TRK		1	1	1	1	1	1996.227	1996.227									
STRL005.T91.EXP.KS00000.D@	WRK000	8598	0340	Y	R	AM	U	2048		TRK		1	1	1		1	1996.220	1999.365									
STRL005.T91.EXP.KS00000.I@	WRK000	8598	0340	Y	R	AM	U	512		TRK		1	1	1		1	1996.220	1999.365									
STRL005.T91.EXP.KS07011.D@	WRK000	8598	0340	Y	R	AM	U	2048		TRK		1	1	1		1	1996.220	1999.365									
STRL005.T91.EXP.KS07011.I@	WRK000	8598	0340	Y	R	AM	U	512		TRK		1	1	1		1	1996.220	1999.365									
STRL005.T91.EXP.KS97001.D@	WRK000	8598	0340	Y	R	AM	U	2048		TRK		1	1	1		1	1996.220	1999.365									
STRL005.T91.EXP.KS97001.I@	WRK000	8598	0340	Y	R	AM	U	512		TRK		1	1	1		1	1996.220	1999.365									
STRL005.T91.EXP.KS98002.D@	WRK000	8598	0340	Y	R	AM	U	2048		TRK		1	1	1		1	1996.220	1999.365									
STRL005.T91.EXP.KS98002.I@	WRK000	8598	0340	Y	R	AM	U	512		TRK		1	1	1		1	1996.220	1999.365									
STRL005.T91.EXP.KS99365.D@	WRK000	8598	0340	Y	R	AM	U	2048		TRK		1	1	1		1	1996.220	1999.365									
STRL005.T91.EXP.KS99365.I@	WRK000	8598	0340	Y	R	AM	U	512		TRK		1	1	1		1	1996.220	1999.365									
STRL005.T91.EXP.PO00000	WRK000	8598	0340	Y	F	PO	FB	3120	80	TRK		1	1	1		1	1996.220										
STRL005.T91.EXP.PO99365	WRK000	8598	0340	Y	F	PO	FB	3120	80	TRK		1	1	1		1	1996.220	1999.365									
STRL005.T91.JUNK1.ESDS.D@	WRK000	8598	0340	Y	R	AM	U	4096		CYL		5	1	75		1	1996.219	1999.365									
STRL005.T91.JUNK1.KSDS.D@	WRK000	8598	0340	Y	R	AM	U	4096		CYL		5	1	75		1	1996.219	1999.365									
STRL005.T91.JUNK1.KSDS.I@	WRK000	8598	0340	Y	R	AM	U	2048		TRK		1	1	1		1	1996.219	1999.365									
STRL005.T91.JUNK1.RRDS.D@	WRK000	8598	0340	Y	R	AM	U	4096		CYL		5	1	75		1	1996.219	1999.365									
STRL005.T91.OUTPUT.FILES	WRK000	8598	0340	Y	F	DA	F	23476		CYL		1		15		1	1996.220										
STRL005.T91.PDS	WRK000	8598	0340	Y	F	PO	FB	9040	80	CYL		2	1	30	25	1	1996.219										
STRL005.T91.PDS.COPY2	WRK000	8598	0340	Y	F	PO	FB	9040	80	CYL		2	1	30	25	1	1996.219										
STRL005.T91.PS	WRK000	8598	0340	Y	F	PS	FB	9040	80	CYL		2	1	30	26	1	1996.219										
STRL005.VK.DATA	WRK000	8598	0340	Y	R	AM	U	4096		CYL		1	1	15		1	1996.221	1999.365									
STRL005.VK.INDEX	WRK000	8598	0340	Y	R	AM	U	2048		TRK		5	1	5		1	1996.221	1999.365									
TOTAL DATA SETS												78	TOTAL ALLOCATED TRACKS												2105		
												TOTAL IDLE TRACKS												349			

Figure 4-24. Sample SVDICT Report

Chapter 5. BACKUP/ARCHIVE

One of the objectives of SAMS:Disk is to provide comprehensive, efficient, common processing of mixtures of non-VSAM data sets and VSAM clusters. Within a single execution you may back up, archive, delete and expire partitioned (PO), partitioned extended (POE), physical sequential (PS), direct access (DA), and OS CVOL catalog non-VSAM data sets. You may also back up, archive, delete and expire VSAM clusters, and back up DMF/EF Type-2 catalogs.

You may do the backups and archives in either an immediate mode, or in a deferred mode in which the back up or archive is actually done at a later time.

You may do this based on: scans of VTOCs of disk volumes, scans of DMF/EF Type-2 or VSAM (but not OS CVOL) catalogs, or the results of a catalog superlocate.

You also have a rich set of criteria to select data sets for and exclude data sets from processing as desired, and a rich set of action command parameters to let you control exactly what you want to do.

Commands Available:	JCL Proc Executed :
ARCHIVE	ARCHIVE
ARCHIVE	DMS
BACKUP	DMS
DARCHIVE	ARCHIVE
(dispose)	ARCHIVE with overridden EXEC parm
DERASE	DERASE
DELETE	DMS
EXCLUDE	DMS
EXPIRE	DMS
FIND	DMS
LISTREQ	LISTREQ
PASSWORD	DMS
SCAN	DMS
SELECT	DMS
SET	DMS

General Information

These functions provide the backup, archive and delete capabilities for non-VSAM data sets, VSAM clusters and DMF/EF Type-2 catalogs at the data set level. For information on volume-level backup and recovery, see the VBACKUP/VRECOVER section beginning on page 251.

Under your control, the SAMS:Disk archives may be on tape, cartridge or disk.

Terminology

Backup

Backing up (copying) data from your online system to the SAMS:Disk archives, leaving the original data set in place.

Incremental Backup

The process of backing up only those data sets that have been modified since the previous backup. This type of backup runs faster than a full volume backup, and does not needlessly use space in the SAMS:Disk archives for redundant copies of data sets that have not changed.

Archive

Backing up data sets to the SAMS:Disk archives, then deleting the original. This is equivalent to “moving” or “migrating” the data set from your online system to the SAMS:Disk archives. From the SAMS:Disk archives, data sets may be “restored” or “recalled” to your online system, or data sets may eventually be “expired” or “rolled off” the archives.

Delete

Refers to deleting the original data set without making a backup copy in the SAMS:Disk archives.

Expire

Refers to deleting an expired data set without making a backup copy in the SAMS:Disk archives. The expiration date can reside in any of the following areas:

- Format-1 DSCB for non-VSAM
- Catalog entry for VSAM

These concepts are implemented with the BACKUP, ARCHIVE, DELETE, and EXPIRE commands described below. These commands in turn run under the umbrella of DSCL, documented beginning on page 23. Depending on how you specify

selection criteria and the action commands, you can use these backup/archive/delete/expire features to perform these and other data storage management tasks:

- Back up full volumes of data sets.
- Back up changed data sets.
- Move unused data sets to the SAMS:Disk archives on tape or disk.
- Move old versions of GDG data sets to the archives.
- Delete old system temporary data sets from work volumes.
- Delete uncataloged data sets.
- Delete unused data sets.
- Delete expired data sets.
- Verify that data sets are on authorized volumes.

BACKUP and ARCHIVE Considerations

Change Bit

The “change bit”, formally referred to as the DS1IND02 field in the format-1 DSCB, supports each non-VSAM data set. The SAMS:Disk SVC recognizes and supports this same change bit. The SAMS:Disk SVC intercepts processing for a data set that is being opened for output and turn on the change bit. The SAMS:Disk incremental backup function identifies the modified data sets by this bit in each VTOC entry. As each changed data set is identified, SAMS:Disk creates and indexes a backup copy, and then optionally turns the change bit off.

Midnight “Rapid Aging” Problems

When SAMS:Disk jobs that examine date fields are run very shortly after midnight, you must take into account the “rapid aging” that occurs at the stroke of midnight. Data sets that may have been created or used merely minutes ago suddenly appear as if the action took place a day ago.

When batch jobs create data sets, the create date used by the operating system is the date the job started, not the date that each specific step of the job started. Hence data sets actually created in steps running after midnight, but part of a multistep job that began before midnight, will have create dates of the previous day—and immediately appear to be one day old!

Any selection criteria based upon date values should take these two items into account, or you may end up selecting many more data sets for action than you intended.

Exempting Data Sets

Data set names or data set name patterns that are to be exempted from selection should be specified by an EXCLUDE command. (See the “*EXCLUDE Command*” topic on page 69 for the command format.)

Exempting Restored Data Sets

If your installation runs archives regularly, you may find that data sets that were restored recently are being archived again. This is because the same selection criteria that caused a data set’s original archival may cause it to be re-archived after it has been restored, if the data set is not modified before the next archive run.

Normally the condition causing the data set’s archival is changed by the requestor, either when it is restored or very shortly thereafter. However, you may also specify an installation option that allows a grace period to be given to any restored data set before it will be re-archived. For more information, see the sysparm descriptions for *RETEXCLD* (page 162) and *RESRETPD* (page 161), both in the *Systems Guide*.

Note: This grace period does not apply to backups, since they do not delete the data set from disk.

Backup of Volume VTOCS

If you specify the SELECT VTOCS and BACKUP command, SAMS:Disk will back up VTOCs from all volumes that are scanned, regardless of whether or not any data sets from these volume are selected for processing. A backup copy of a VTOC may subsequently be used by RECOVER processing to restrict the recovery to only those data sets listed in the backup copy of the VTOC, excluding from recovery those data sets that were deleted from the volume prior to the backup run.

The index for an indexed VTOC is not backed up by SAMS:Disk because it is not needed. It contains no useful or helpful data for either the archive or the restore function. The entries in the index for a VTOC are automatically rebuilt at restore time due to the standard allocation interfaces used by SAMS:Disk.

The following restrictions apply to this support:

- Only the most recent copy of a volume's VTOC found in the SAMS:Disk archives will be used as the screening list during recovery processing.
- The VTOC copy cannot be used to overlay the current VTOC on a volume. SAMS:Disk dynamically rebuilds each VTOC entry as each data set is restored.
- If you have a SAMS:Disk Security Interface installed, you will need access to the data set name "VTOC.volser", where "volser" is the volume on which the VTOC resides. Both BACKUP and IXMAINT functions will query any SAMS:Disk Security Interfaces for authority to process this name.

Task Performance

You may execute multiple backup or archive jobs concurrently on the same or different CPUs. By default, SAMS:Disk will generate unique archive data set names during dynamic allocation of the ARCHIVE0 and ARCHIVEC dd statements. This prevents the different jobs from waiting for a tape data set name enqueue held by another job.

If you provide the ARCHIVE0 or ARCHIVEC dd statements in your JCL and specify sysparm ARCTNAME with a value of "D", SAMS:Disk will append a 22-character date/time stamp to the name specified in the JCL. Refer to the topic "*Naming the Archives*" on page 59 of the *Systems Guide* for details.

Each backup or archive job should also specify a list of volumes to be processed that is different from the other jobs. To reduce channel contention among the jobs, the volume lists should accomplish as much channel separation as possible.

Task Restart Considerations

After each DASD data set is written to tape successfully, a SAMS:Disk index entry is placed in the DSNINDEX subfile and the delete and catalog actions are taken as needed. With this design, any tape created by SAMS:Disk is a good tape, even though a system failure or job cancel may have occurred.

If a job fails, the exact same archive or incremental backup command request may be resubmitted. SAMS:Disk will request a new scratch tape and processing will continue where it left off. In reality, all volumes and all data sets are actually rescanned for selection, but if they were successfully selected and processed in the immediately preceding partial run, most selection types will not reselect them. For example, an unused data set that was archived and deleted will not be seen by the restarted job. Changed data sets backed up successfully will no longer have the change bit on and will therefore not be reselected, unless the NORESET parameter is also specified.

If an archive or backup fails because the data set is in use, the data set can be automatically queued for deferred processing by specifying sysparm *ARCFDEFR* with a value of Y. For more information on this sysparm, turn to page 118 in the *Systems Guide*.

ARCHIVE Report Sequencing Options

The archive report can be produced in three sequences:

- data set name sequence
- data set name sequence within volume
- exact processing sequence (unsorted)

By default, all three reports are produced. See sysparm *ARCHSORT* on page 119 of the *Systems Guide* if you want to change the defaults. Any combination of the three report sequences may be specified.

Data Set Compression and Reorganization

When a data set is being read from disk, SAMS:Disk processes the data blocks within it “sequentially” as dictated by the organization type. Thus only active and used data blocks are read. This results in PDS and VSAM data sets being copied to the archive tape in compressed format. Any subsequent restore or volume recovery results not only in the consolidation of several extents into one (defragmentation), but also in the compression/reorganization of the data within the single extent (eliminating the additional need to compress and reorganize these same data sets).

Non-VSAM Considerations

The following section details special processing options that can be used to tailor the non-VSAM backup and archival functions.

Backup of SYSCTLG

SAMS:Disk will back up OS CVOL catalogs (DSN=SYSCTLG), but you will probably need to specify sysparm ARCDSORG with a value of Y. This is because most SYSCTLG data sets have no DSORG indicated in the VTOC DSCB. Due to the embedded TTR values, SYSCTLG data sets must be restored to the same device type from which they were archived. (The Move function, described beginning on page 331, has the ability to move these data sets to different device types, updating the internal TTR pointers.)

PDS Anomalies

SAMS:Disk does extensive testing for errors and inconsistencies in partitioned data sets, and provides installation options for handling PDSs in which logical errors have been detected. Standard default processing is to issue diagnostic messages for the errors, and then bypass them. Consult with your local SAMS:Disk installation and support staff to determine what options are in effect if any of your PDS data sets are diagnosed to have anomalies.

This facility provides an automated means of deciding which data sets on disk should be backed up or archived, and which should be bypassed. For related information, please turn to the *"Processing PDSs that Contain Anomalies"* on page 69 of the *Systems Guide*.

Model DSCB Processing

Data sets with zero extents and no secondary allocation amount are most commonly used as model DSCBs. They may be identified with the DSCL selection criteria MODELDSCL.

Since model DSCBs are not opened and used like other data sets, their modify bit and last used date are never modified. Because they take up no space on the volume, by default SAMS:Disk will not archive model DSCBs unless you specify sysparm ARCMODEL with a value of Y. Regardless of the setting of ARCMODEL, SAMS:Disk backs up and restores model DSCBs.

VSAM Considerations

The following section details special processing options that can be used to tailor the VSAM backup and archival functions.

Verify of VSAM Clusters

While attempting to open a cluster for input, SAMS:Disk may receive a failure from open processing indicating that the data set was not properly closed by a previous task. When this error is encountered, an JSCVSUT VERIFY is required to synchronize the end-of-file marker in the data set with the value found in the catalog. By specifying sysparm VSVERIFY with a value of Y, SAMS:Disk will automatically issue a verify when this condition is encountered and will then proceed to reopen the data set and continue archival processing.

DFP automatically issues a verify against a cluster if the need is detected at open time. Since this feature is viewed as an integrity exposure to some, the user can turn this feature off for SAMS:Disk processing by specifying sysparm VSSYSVER with a value of N (the default). Note that if this feature is turned off, SAMS:Disk can still be instructed to issue the verify by specification of sysparm VSVERIFY with a value of Y.

Backing Up or Archiving Empty Clusters

SAMS:Disk will archive any valid cluster that passes selection testing, regardless of whether or not it contains data. If the cluster is empty, only its attributes are copied to tape.

SAMS:Disk selects a VSAM cluster for Archive/Backup processing if the volume serial for any component or extent of the cluster meets the volume selection criteria. This occurs even if other components of the cluster do not meet the volume selection criteria.

The cluster-level selection criteria must also be met before the cluster is selected for processing. However, SAMS:Disk will not select a VSAM data set for Archive/Backup processing more than once within the same job step.

The files data set contains the volume serial of the first data component of a VSAM cluster. During Archive/Backup processing, SAMS:Disk issues a message that shows this volume serial. The volume serial must be correctly specified for a RECOVER run that uses the VOL= parameter. SAMS:Disk can only RECOVER clusters if the first data component volume is selected.

Never Opened Clusters

A never opened VSAM cluster is one which has been defined but has never been opened for output. These are empty clusters because their VSAM TIMESTAMP in their catalog entry and their last use date in the format-1 DSCB are both zero.

By default, these clusters will not be selected for backup or archive. If you want to select these clusters for processing, you can do so by specifying sysparm VSUNOPEN with a value of "Y".

With VSUNOPEN set to "Y", SAMS:Disk compares CREDIT to ARCDT. If the CREDIT is greater than or equal to the ARCDT, SAMS:Disk will select these clusters with the criteria MODIFIED,EQ,YES.

There is one exception, and that occurs on the day the cluster is defined. On that day, SAMS:Disk will select "defined, but never opened" clusters every time the criteria (MODIFIED,EQ,YES) is executed.

For more information, please review the sysparm description for *VSUNOPEN* on page 193 of the *Systems Guide*.

Processing Clusters with Alternate Indexes

SAMS:Disk will automatically retrieve all of the information necessary to recreate a base cluster's alternate index(es) and/or path(s), and will archive the information with the base cluster. In addition, the alternate index can be processed as a separate entry from the cluster (which is referred to as a physical alternate index). If you intend to process physical alternate indexes, please review the topic "*Alternate Index Support*" on page 175, and "*Restoring Alternate Indexes*" on page 218.

Deleting Unexpired Data Sets

By specifying sysparm VSMPURGE with a value of Y (the default value), SAMS:Disk will allow the deletion of unexpired data sets. This is equivalent to using the PURGE parameter on an JSCVSUT DELETE request.

Processing Clusters in Use by Other Tasks

Normal SAMS:Disk processing determines the disposition type to use for data set allocation based on the share options with which a cluster is defined. If a cluster is defined with SHAREOPTIONS (1 3), it is allocated with DISP=SHR; otherwise it is allocated with DISP=OLD. By specifying sysparm VSALOSHR with a value of Y, all allocations will be made with DISP=SHR. In addition, the user exit USERVENQ can be invoked to determine the allocation desired on the data set level.

Note: Since SAMS:Disk does not issue ENQs for VSAM data sets, it cannot be held responsible for the integrity of a data set that was archived with a disposition type other than that which it would have selected. SAMS:Disk relies solely on the integrity features offered by VSAM through its share options.

Recataloging for VSAM Auto-Restore

When a VSAM cluster is archived and recataloged to the SAMS:Disk pseudo-volume, alias entries are defined for each of the alternate index and path names as well as the data and index component names. This enables the auto-restore function to be invoked when you reference either the base cluster, a component name, an alternate index, or a path name.

Restore processing removes the catalog entry to the pseudo-volume and all of the aliases associated with it, and redefines the correct VSAM catalog entries. Should the restore process fail, SAMS:Disk attempts to recreate the initial status by recataloging the cluster to the pseudo-volume and associating the proper alias names to it.

VSAM Backup and Archive Restrictions

KSDS clusters that use the KEY-RANGE attribute are processed, but if the data set was archived prior to Release 7.8 of SAMS:Disk, the KEY-RANGE attribute is not retained when restored.

Clusters may span up to 20 volumes.

VSAM and DMF/EF Type-2 catalogs cannot be archived. However, backup copies of DMF/EF Type-2 catalogs can be taken.

Relative Record Data Sets (RRDS) that contain more than 16,776,000 active slots cannot be archived.

If the proper catalog cannot be located via standard VOS3 catalog searching orders, the //STEP CAT dd statement must be provided.

Access Methods for VSAM Clusters

During backup and restore processing, SAMS:Disk processes VSAM data at the cluster level rather than the data space level. Logically, data processing accesses records within a file. Physically, the file is a group of areas (extents) on one or more DASD volumes. For VSAM, the cluster entity is equivalent to the logical file, while the data space is equivalent to the physical storage. Processing data at the physical (data space) level makes the data dependent on both the storage medium (device type) and the method of access.

An archive system may retain data offline for an extended period of time and when restoration of the data file becomes necessary, the storage medium may no longer be available. Also, in the event of a disaster, the recovery site may not have the same physical medium or access method available. Processing data at the logical file level is one way of providing data independence; that is, the capability to restore the logical data to a different physical medium or possibly using an alternative method of access. In addition, logical processing allows the data to be reorganized as a byproduct of processing.

SAMS:Disk uses normal access methods to read VSAM clusters. It can read them in one of two ways:

1. Logical record processing
2. Control interval processing

The method SAMS:Disk will use is controlled by sysparm VSACCESS. The basic difference in the methods is that with logical record processing, SAMS:Disk issues requests for individual data records—it is up to VSAM to read the control intervals and perform buffer management functions. With control interval access, SAMS:Disk requests specific control intervals that it must, in turn, deblock into their component logical records. This deblocking process can be suppressed, however, when necessary (such as when processing XDM/RD data bases). More on this later.

With control interval access, SAMS:Disk performs its own buffer management. The size of the buffer area is equal to the size of one data component Control Area size.

SAMS:Disk will read all Control Intervals sequentially within a Control Area and store them before processing the logical records within the Control Intervals. This technique of reading the entire Control Area is done to take advantage of the read ahead functions of VSAM. By reading the Control Intervals sequentially, VSAM will automatically read the entire Control Area when the first read is issued. The result is a single read I/O for each Control Area processed.

Alternate Index Support

There are two methods SAMS:Disk may use to archive and restore alternate indexes. The first method is always performed and cannot be overridden: the catalog information required to redefine an alternate index is always archived with its base cluster. When the base cluster is restored, the alternate index can be redefined from this information. In this case, an entry is made in the SAMS:Disk files data set for only the base cluster and not its alternate index(es).

The second method of archiving an alternate index is to explicitly name it on an archive command with sysparm VSARCAIX specified with a value of Y, or to run a backup of the base cluster with sysparm VSARCAIX specified with a value of Y. This method causes the alternate index to be physically processed as a data set; as such it will have its own entry in the SAMS:Disk files data set and the data records from the alternate index will be read and archived along with the catalog information. Using this latter method, the cluster index will be referred to in this section as the "physical" alternate index.

If a disposition of DELETE is specified or assumed on the command, alternate indexes will not be archived. This is to ensure the alternate index will not be deleted before the base cluster is processed, which would remove the association from the base cluster.

The alternate index can always be redefined and built based on the catalog information archived with the base cluster. There are special processing considerations required to restore a physical alternate index, and they are detailed later in this section of the manual. Prior to reading the sections on restoring VSAM clusters and alternate indexes, make sure you understand the difference between restoring a physical alternate index as opposed to rebuilding the alternate index from catalog information saved with the base cluster.

It should be noted that clusters that were archived with a release of SAMS:Disk prior to Release 7.5 will not have the alternate index information with the base cluster. It is therefore necessary to redefine the alternate indexes of these clusters manually after the base cluster is restored by SAMS:Disk.

DMF/EF Type-2 Catalog Backup and Recovery

Backup and recovery of catalogs is a complex and critical procedure. To assume that it can be reduced to a process that is as simple as “pushing a button” is to make a serious mistake! The myriad of dependencies among the operating system, the type of catalog (OS CVOL, VSAM user catalog, or DMF/EF Type-2 catalog), and the data related to the catalog entries almost always demands unique recovery steps to obtain a catalog that is truly useful.

Recovering a catalog from a backup copy can be meaningful or useful for both DMF/EF Type-2 catalogs and OS CVOLS, but rarely (if ever) for VSAM user catalogs. The old VSAM user catalogs have many dependencies such as time stamps and volume ownership that make it nearly impossible to use them if they are recovered independently from all of their clusters. The ability (JSCVSUT functions) to re-synchronize them is just not available. Hitachi addressed these concerns by providing DMF/EF Type-2 catalogs, and the enhanced abilities in JSCVSUT to re-synchronize them. For these reasons, SAMS:Disk provides the ability to automate and extend the catalog backup process for DMF/EF Type-2 catalogs, but you must provide the knowledge and analysis to see that the correct recovery steps are taken—steps that suit your particular case and achieve the results you desire.

Backing up DMF/EF Type-2 catalogs can be automated simply by specifying `SELECT CATALOGS=` in front of a `BACKUP` command, or by adding the `BACKUP` parameter to the `SCAN CATALOGS=` command. If a `SELECT` statement is not specified with the `SCAN CATALOGS=` command, SAMS:Disk will backup DMF/EF Type-2 catalogs. As each catalog is selected for processing, SAMS:Disk links to the JSCVSUT `EXPORT` function, intercepts the records, and writes them in SAMS:Disk format to the archive media. To extend or enhance the backup and recovery process, the `ALIAS` entries for the catalog are also extracted from the master catalog and appended to the exported catalog data. This provides SAMS:Disk with the ability to dynamically redefine them in the master catalog per the `DEFALIAS=YES/no` parameter on the `RECOVER` command. Without this ability, all alias pointers in the master catalog must be rebuilt manually because they are deleted (lost) when a catalog is imported.

Parameter `EXPORTF=NO/yes` is also available for the `RECOVER` command. Specifying `YES` instructs `SAMS:Disk` not to `IMPORT` the catalog, but to create a sequential data set on disk that can be used as direct input to a native `IMPORT`; that is, the data set will be in “export format” that `IMPORT` can use. When running with this option, messages are issued to identify the `ALIAS` entries that you may need to redefine in the master catalog after you actually do the `IMPORT`. (As mentioned above, a backup copy and an `IMPORT` done under DFP 2.3 or above should preserve the `ALIAS` entries. For earlier versions of DFP you will need to redefine them manually.)

It should be evident from the descriptions above that `SAMS:Disk` automates and extends the use of Hitachi’s `JSCVSUT` `EXPORT` and `IMPORT` functions to provide for much easier backup and recovery of `DMF/EF` Type-2 catalogs. It is not the intent of this documentation, however, to present a tutorial or guideline regarding a multitude of recovery scenarios, the problems encountered in each, and the steps necessary to recover successfully. Other publications or classes specifically designed to address the complexity of these topics should be consulted if needed. Your analysis and planning should be based upon the abilities provided by `EXPORT` and `IMPORT` (as automated and extended by `SAMS:Disk`) and the other `DMF/EF` Type-2 resynchronization functions within `JSCVSUT`.

If a catalog being recovered still exists on disk, `JSCVSUT` will overlay it only if its temporary export flag is on. Otherwise the existing catalog must be deleted before the `IMPORT` can be done. See sysparm `CATBKDEL`, described on page 125 of the *Systems Guide*, for an option to have `SAMS:Disk` attempt this delete automatically for you. (A “`DELETE usercat UCAT RECOVERY`” command will be issued.)

If a catalog being recovered does not exist on disk, but its entry in the master catalog still exists (for example, the volume containing the catalog has been reinitialized due to a hardware failure), you will probably need to issue an `JSCVSUT` “`EXPORT usercat/masterpassword DISCONNECT`” command before the recovery will succeed.

One very important planning item to consider is the following. Depending upon which catalog is lost, the ability to submit and initiate jobs may be impossible without hardcoding volume information in the `JCL` (because the catalog normally used to provide this information is the one that was lost). One of these jobs may be the `SAMS:Disk` recovery run, or perhaps the `JSCVSUT` `IMPORT` job! Being prepared to provide the volumes in the `JCL` for either of these jobs can be essential!

`JSCVSUT` can dynamically allocate each `DMF/EF` Type-2 catalog as it begins the `EXPORT` function, but it does so with `DISP=OLD` and does not deallocate (free) the catalog when it is done (it waits for step termination to do so). To avoid what could be a lengthy tie-up, `SAMS:Disk` uses its own dynamic allocation to allocate each needed catalog, and then deallocates (frees) it immediately after the `EXPORT`. For importing a catalog, however, `SAMS:Disk` lets `JSCVSUT` perform its own allocation.

Before recovering an DMF/EF Type-2 catalog and its associated clusters, review the value specified for sysparm VSPREDEF carefully. The default value of N prevents SAMS:Disk from overwriting preallocated (or in this case, pre-existing) VSAM clusters during restore processing. If it is specified with a value of Y (not recommended), SAMS:Disk will overwrite the existing clusters with the REUSE attribute. The best method would be to leave VSPREDEF with the default value (N). Recover the DMF/EF Type-2 catalog and the clusters. Then restore the exceptions (if necessary) using the SCRATCH parameter of Restore.

Processing Clusters that have Non-Standard Formats

Certain program products use VSAM but have a non-standard internal format in the VSAM control interval. Among those known to have such a problem are Hitachi's XDM/RD database management system, Hitachi's linear data sets (SVSAM) and certain data sets created by MSA products. These data sets must be processed in a special manner by SAMS:Disk.

The typical control interval in the data component of a VSAM cluster looks something like this:

- Record n = data record (actual user data)
- Free space = space that is unused in the control interval
- RDF = Record Descriptor Field. There may be one for each data record in the control interval, but this is not always the case.
- CIDF = Control Interval Descriptor Field. This field normally describes the amount of space being used by the data records portion, as well as the number of free space bytes remaining.

When SAMS:Disk processes a VSAM data set, it can process in one of two modes: logical record or with control interval access. With logical record access, SAMS:Disk issues requests for individual data records and will never see the RDF and CIDF fields. With control interval access, SAMS:Disk is returned the entire control interval with each read request; it must "deblock" the control interval using the CIDF and RDF control information.

The problem with these non-standard data sets is that they do not maintain the CIDF and/or RDF records, or they use their own techniques in managing the internal data. When SAMS:Disk (or JSCVSUT, for that matter!) tries to process these records using logical record techniques, unpredictable results will occur. There is no way of knowing that the CIDF and RDF are not present, or that they have assumed a different meaning from what the initial design of VSAM intended.

The way to avoid these problems is to read the data set with control interval access (as SAMS:Disk has been able to do for some time) and then archive the entire control interval as a single record. In this way SAMS:Disk can ignore what should be the CIDF and RDF records; it will not attempt to interpret the data inside the

control interval. The only disadvantage to this technique is that at restore time the control interval size cannot change. This means that SAMS:Disk must put the cluster back exactly the way it archived it.

By default, SAMS:Disk will always write archived data sets in logical record format. SAMS:Disk automatically detects data sets that do not observe logical record definitions, such as XDM/RD data sets, Oracle data sets and some data sets created by MSA products. When one of these data sets is found, SAMS:Disk will automatically shift to control interval image copy format when archiving the data set. If you have VSAM data sets in non-standard formats that SAMS:Disk can not detect, you have two options for processing these data sets. You can code sysparm VSARCFMT with a value of C, which will cause all VSAM ESDS data sets to be processed in control interval image copy format, or you can code your own user exit called VSACESEX to selectively determine which mode of archiving should be used for each data set.

At this time, entry-sequenced data sets (ESDS) are the only type supported by control interval image copy format. KSDS and RRDS clusters will always be archived in logical record format.

Incremental Backups

General Information

The SAMS:Disk incremental backup function is implemented as a selection criteria MODIFIED under DSCL.

Volume Level Full Backups

Your installation may also want to include volume-level backup runs as part of its backup plan. This function is described in the *VBACKUP/VRECOVER* section, beginning on page 251.

SAMS:Disk volume-level backup or recover runs do not use the traditional data set-level processing technique that is used by normal DSCL backup and archive processing. Therefore, running the volume-level backup will take much less time than the full volume backup method described earlier in this section. However, because *VBACKUP* and *VRECOVER* functions are volume-oriented, you cannot use the volume-level backup tape to restore individual data sets.

Therefore, you may wish to use the volume-level backup as your primary backup protection only for volumes with a high activity level; that is, most of the data sets are changed every day or every week. This will save time when running the daily or weekly backups.

If you need to restore data sets on this type of volume, you can issue the *VRECOVER* command with the *UPDATE* parameter specified, which will restore the entire volume, then update the modified data sets from your incremental backups. Turn to the *VBACKUP/VRECOVER* section on page 251 for more details.

Implementing Incremental Backup

The following questions and answers represent some of the typical concerns that need to be addressed in order to implement incremental backup.

1. How do I get an initial copy of all my data sets?

By using SAMS:Disk to produce a full data set backup. This is done by running:

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
PASSWORD MASTER=the-SAMS:Disk-master-password  
SCAN REALVOLS  
SELECT VTOCS  
SELECT CATALOGS=/  
SELECT DS NAMES=/  
BACKUP
```

2. How do I establish proper retention periods for the archive/backup volumes, and all of the data sets contained on those volumes?

First, recall how retention periods are assigned by SAMS:Disk for non-VSAM data sets. As each data set is copied to the archives (on tape or disk), an expiration date is assigned to its index entry. The value of the expiration date is determined as follows: A default expiration date is calculated using the default retention period value of sysparm RETRETPD, or from the RETPD or EXPDT parameter if it is present on the BACKUP command.

In addition to each archived copy of a data set being assigned its own retention period, the containing archive volume (the tape, or a sequential data set on disk) is also assigned an expiration date. SAMS:Disk default processing assigns the “never scratch” Julian date of “99365” to each of the archive volumes. This ensures that they will not be written over until all contained archived data sets have expired. The DSNDELETE function of index maintenance will return tape volumes to scratch status (and delete backup or archive data sets kept on disk) when it determines that all contained data sets have indeed expired.

You may assign an alternate expiration date for archive volumes that are dynamically allocated by specifying sysparm DYNEXPDT with either an expiration date or retention period value. You may also provide the //ARCHIVE0 or //ARCHIVEC dd statement in your JCL. The value you specify for the RETPD or EXPDT parameter will be assigned to the archive volume.

With this in mind, it’s appropriate to decide what the default retention period should be — 15, 30, or however many days.

3. Since incremental backup creates a new copy only when a data set is modified, and assigns only that copy an expiration date, how can I protect that copy from expiring — and thus being dropped from the archives — if the data set is not modified again sometime within the retention period of the prior copy? (For example, a read-only data set will never have new backup copies generated after its initial backup if backup uses only the change bit selection criteria.)

One technique is to coordinate the retention period assigned to each archived data set with the frequency with which you take full volume backups.

For example, assume we want to ensure that there will always be one backup copy of every data set, and full volume backups should be done only once a week. This can be accomplished by taking a full volume backup one day each week and incremental backup runs every

intervening day. Both the full volume backup and the incremental backup runs should specify RETPD=8 on the BACKUP command. The incremental backup run should also specify MODIFIED,EQ,YES on the SELECT command.

A second and more efficient technique can be used if the SAMS:Disk SVC is being used to record the last modify date in the data set's VTOC entry at the same time the change bit is being turned on.

Using this technique, instead of doing full volume backups once a week (usually on the weekend), SAMS:Disk will select and back up only those data sets that have NOT been modified during the seven-day cycle. This eliminates time wasted in taking redundant copies of data sets that have already been backed up because they changed during the week.

For example, on Monday through Saturday, back up only the data sets being changed (MODIFIED,EQ,YES). On Sunday, back up the changed ones, but also back up any data set NOT changed within the last seven days by using this statement:

```
MODIFIED, EQ, YES, OR, MODDT, LT, TODAY-7
```

To ensure that you always have at least one copy of every data set in the archives, set the RETPD value to 15 days. A data set changed (and backed up) on Monday won't be picked up again by the "unmodified in seven days" test until the second following Sunday.

4. What if I want to keep two backup copies in the archives?

For the previous example, simply specify RETPD=22 on the BACKUP commands.

In general, set the RETPD= parameter to $(X+1)$ times the number of days between the runs that pick up the unmodified data sets, where X is the number of backup copies you wish to ensure. Add 1 to the result to allow a day overlap.

For example, with monthly runs to pick up the unmodified data sets, and ensuring one copy, use RETPD=63 (that is, $(2*31+1)$). To ensure two copies, use RETPD=94 (that is, $(3*31+1)$).

5. Can you draw a picture to demonstrate this?

Consider the simple case of weekly runs to pick up the unmodified data sets, using RETPD=15 to ensure one copy of all data sets. In the

two-week diagram below, C represents the creation of a tape and E its earliest possible expiration (the RETPD value).

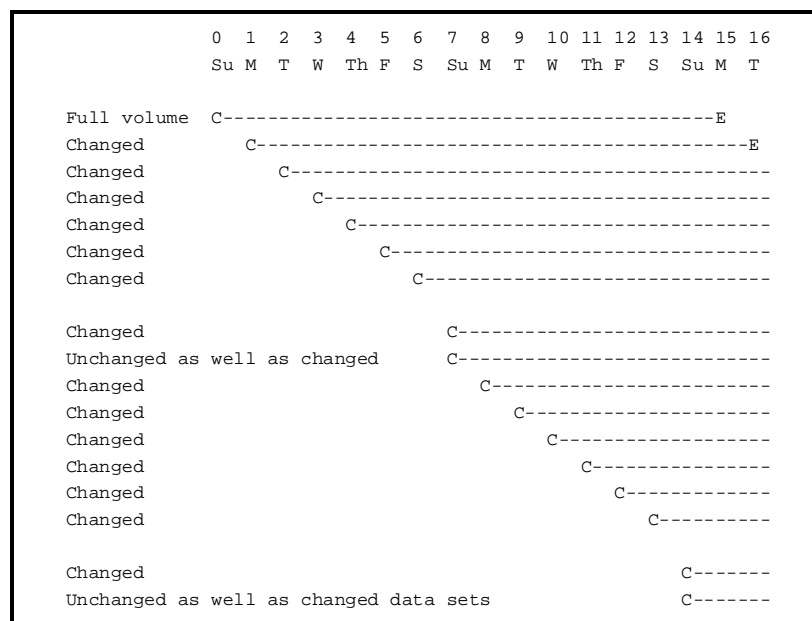


Figure 5-1. Weekly Production Backup Schedule

Notice that if a data set existed on Sunday (day 0) but is never modified, a new copy is generated each successive Sunday, before the previous tape (and version of the data set) has expired. If the data set is modified every day, each incremental backup tape will also contain a copy. At the end of the week, there may be from one to seven copies of the data set, but a minimum of one is ensured at all times. A data set modified on Monday (day 1) will not be backed up again until Sunday (day 14), but the prior copy is being kept until Tuesday (day 16), so one copy is guaranteed.

Also notice that running incremental backup on a daily basis is a somewhat arbitrary factor so far as it concerns the previous objective. Running it twice daily or only twice a week ensures the minimum copy goal as long as the RETPD= value is the same. But, most important, notice that full volume backups are never needed again!

6. How do archive MERGE and index maintenance fit into this scheme?

It depends on the characteristics of your data sets and the archive volumes that are created (be they tape or disk data sets). If all data sets backed up are assigned the same RETPD, the entire archive volume (for example, a tape) will expire at the end of the period. If each tape is also nearly completely filled, deleting the expired data sets is preferred because MERGE will accomplish very little. However, if the percentage of tape used in several cases is quite low, merging them

may indeed be useful. Consult the "*MERGE*" section, beginning on page 313, to gain a better understanding of this capability.

Incremental Backup Restrictions

The following restrictions should be observed when planning your installation's incremental backup plan. These restrictions apply solely to the incremental backup function, not to backup processing in general.

1. A SAMS:Disk Stand-Alone Restore (DMSSAR) function is provided as a separate product. Incremental backup does not support a stand-alone capability, and if you are not using the SAMS:Disk Stand-Alone Restore program, backup copies of data sets needed to IPL the system must be provided by some other means.

The SAMS:Disk Stand-Alone Restore function allows you to recover data sets critical to bringing up your operating system. You must produce a volume-level backup tape before you can use this function. (See the *VBACKUP/VRESTORE* section beginning on page 251, and *Stand-Alone Restore Guide* for further information on this program.)

2. SAMS:Disk enqueues with DISP=SHR on each individual data set being backed up. The operating system or a subsystem may have some data sets enqueued with DISP=OLD for the entire duration that the system/subsystem is running. This will prevent SAMS:Disk from ever backing up these data sets and an alternate means must be used. (Actually, if you wish, you may suppress the SAMS:Disk enqueue via user exits to allow the backup copy to be taken anyway.)

For more information, please review the user exit descriptions for *USERSENQ* and *USERVENQ* beginning on page 256 of the *Systems Guide*.

3. Data sets marked UNMOVABLE are not supported by SAMS:Disk and are automatically bypassed. An alternative backup means must be used. For details, refer to the sysparm description for *UNMOVEAR*, located on page 179 of the *Systems Guide*.

BACKUP Command

The BACKUP command instructs SAMS:Disk to create a backup copy of an online data set, and to index this copy in the SAMS:Disk files data set. Only those data sets satisfying the DSCL selection test(s) will be subject to SAMS:Disk processing.

Overview

The backup function may be simulated by using the MODE parameter on the DSCL SET command, described on page 35.

Deferred Requests

Requests to back up a data set may be deferred over a period of time until a batch DISPOSE run can be performed to process all requests deferred since the previous batch run. Use of this facility is desirable, in that a single backup tape mount is required instead of individual mounts for each data set.

A deferred request is made by adding the DEFER parameter, described on page 187, to the BACKUP command.

Deferred backup requests are queued and kept in the ARCHCMDS subfile of the SAMS:Disk files data set. Once a queued request has been processed, its completion status is maintained for a default period of five days, after which time it is deleted automatically during the next DISPOSE run. The output from the DISPOSE run or a batch LISTREQ must be inspected to determine the outcome of processing for the request. Sysparm *DARCKEEP*, described on page 127 of the *Systems Guide*, may be used to alter the default period for which the status is maintained in the ARCHCMDS file.

Member Level Backup

A subset of a partitioned data set may be designated by specifying a list of one or more of its members or member alias names. Associated member and alias names are always maintained, regardless of which are specified in the list.

A member level backup is requested by adding the MEMBER= parameter, described on page 187, to the BACKUP command.

Restrictions

Deferred Backup

A subset of a PDS (specified by the MEMBER= parameter) is not supported.

Syntax

The syntax of the BACKUP command is as follows:

```
BACKUP RETPD= , EXPDT= , ACCESS= , RPT=CIA , CIAOPT= , NORESET ,
      DEFER , NEWNAME= , NEWHLQ= , MEMBER=
```

RETPD=

By default, data sets are assigned a retention period equal to the value of sysparm RETRETPD. To override the default, specify a one- to five-digit number indicating the number of days the selected data sets are to be retained in the SAMS:Disk archives. The date calculated from this retention period cannot exceed the year 2155. For related information, review the sysparm description of EXPDTUSE on page 143 in the *Systems Guide*.

If multiple BACKUP (and ARCHIVE) commands are entered, each may have a different RETPD= or EXPDT= value. You may not specify both the RETPD and EXPDT parameters.

EXPDT=

By default, data sets are assigned a retention period equal to the value of sysparm RETRETPD. To override the default, specify a date constant in any acceptable SAMS:Disk format. For a list of valid dates, turn to the topic *"Date Formats"* on page 8. For related information, review the sysparm description of EXPDTUSE on page 143 in the *Systems Guide*.

SAMS:Disk treats DSNINDEX Julian expiration dates "99000" (except as described in the next paragraph), "99365", "99366" and "99999" as "never scratch" dates. That is, data sets whose ARCHVOLS or DSNINDEX records use these dates as expiration dates will not be deleted automatically as expired by the SAMS:Disk function IXMAINT.

By default, SAMS:Disk treats Julian expiration date "99000" as a never scratch date. But if you set sysparm UNCATDEL to a value of Y, SAMS:Disk treats Julian expiration date "99000" as a "catalog control date." That is, data sets backed up or archived with an expiration date of 99000 will be deleted automatically by the SAMS:Disk function IXMAINT when the data sets become uncataloged. What this means is, if your catalog goes bad or someone uncatalogs some data sets, all back-ups and archives of these data sets will be deleted on the next IXMAINT run. Although you may use EXPDT=99000 for any purpose, it was intended for use with archiving generation data groups (GDGs), the generations of which are managed by the catalog. For example, to archive all GDG generations, leaving the two most recent on disk, deleting generations that roll off of the catalog, specify sysparm UNCATDEL with a value of Y and:

```
SCAN REALVOLS
SELECT CRITERIA=(GDGGEN,LE,- 2)
ARCHIVE DISP=RECAT,EXPDT=99000
```

If multiple BACKUP (and ARCHIVE) commands are entered, each may have a different RETPD= or EXPDT= value.

You may not specify both the RETPD and EXPDT parameters.

ACCESS=

You may override the value of sysparm VSACCESS for this BACKUP command by specifying ACCESS= with a value of C for Control Interval (CI-level) access, or L for logical-level access.

RPT=CIA

This option allows the Cluster Internal Analysis (CIA) report to be produced as a by-product of live mode backup of VSAM clusters. The CIA report will not be produced during simulate runs. Refer to the description of this report on page 103 for further information.

CIAOPT=

If the RPT=CIA parameter is specified, this parameter can be used to specify which options are to be used. Refer to the description of this option on page 103 for further information.

NORESET

If a data set is backed up, normal processing will reset (to zero) the change bit DS1IND02 in the format-1 DSCB. Specification of this parameter will cause processing to bypass updating the change bit. This feature would be useful for full-volume backup runs for off-site storage, where you do not want to prevent the data set from being processed by the next incremental backups.

NEWNAME=

An alternative data set name to be assigned to the DSNINDEX record for the backed up data set (a relative GDG name is also permitted). The backed up version is indexed under this name, which may be up to 44 characters in length.

DEFER

This parameter may be specified on the BACKUP command to indicate that the request is to be queued for later processing. For an overview of this function, turn to the topic *"Deferred Request"* on page 185.

NEWHLQ=

An alternative high level qualifier can be assigned to each data set selected for backup (a relative GDG name is also permitted). The backed up version is indexed in the files data set under this name, which may be up to 44 characters in length.

MEMBER=

To back up a subset of a partitioned data set, this parameter is used to specify one or more member and/or pattern names to constitute the backed up version of the

data set. A valid pattern for member level processing is any character followed by the '/' pattern character. This is the only valid pattern character available for this parameter. As an example:

```
MEMBER=(A/,GEN/)
```

For detailed information on the '/' pattern character, please turn to its description on page 6.

A maximum of 25 entries is supported and is only valid when an immediate backup is being done (not a deferred backup).

For an overview of this function, turn to the topic "*Member Level Backup*" on page 185.

BACKUP Command Examples

Example 1

Backup for seven days the critical data sets LABS.DMSNEXT.INSTALOG.

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
FIND DSN=LABS.DMSNEXT.INSTALOG  
BACKUP RETPD=7
```

Example 2

Backup for seven days all of the critical data sets LABS.DMSNEXT.SOURCE, LABS.DMSNEXT.LOADLIB, LABS.DMSNEXT.ASSEMSRC and LABS.DMSNEXT.INSTALOG

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
FIND DSN=LABS.DMSNEXT.SOURCE  
FIND DSN=LABS.DMSNEXT.LOADLIB  
FIND DSN=LABS.DMSNEXT.ASSEMSRC  
FIND DSN=LABS.DMSNEXT.INSTALOG  
BACKUP RETPD=7
```

Example 3

Backup for seven days all of the critical data sets LABS.DMSNEXT.anything.

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
FIND DSN=LABS.DMSNEXT./  
BACKUP RETPD=7
```

Example 4

Backup for 28 days, newly modified non-VSAM data sets and VSAM clusters, plus all non-VSAM data sets and VSAM clusters not backed up in 27 days, plus all

XDM/RD database components (which do have a valid last used date), plus all DMF/EF Type-2 catalogs and all VTOCS.

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
PASSWORD MASTER=the-SAMS:Disk-master-password  
SCAN REALVOLS  
    SELECT CRITERIA=(MODIFIED,EQ,YES)      newly modified  
    SELECT CRITERIA=(ARCDT,LE,TODAY-27)    about to roll off  
    SELECT CATALOGS=/                      my DMF/EF Type-2 catalogs  
    SELECT VTOCS                          for RECOVER VOL=,VTOCS  
    BACKUP RETPD=28
```

ARCHIVE Command

The ARCHIVE command instructs SAMS:Disk to create a backup copy of an on-line data set, to index this copy in the SAMS:Disk files data set, and to dispose of the online copy as requested. Only those data sets satisfying the DSCL selection test(s) will be subject to SAMS:Disk processing.

Overview

The ARCHIVE function may be simulated by using the MODE parameter on the DSCL SET command, described on page 35.

Deferred Requests

Requests for data set archival may be deferred over a period of time until a batch DISPOSE run can be performed to process all requests deferred since the previous batch run. Use of this facility is desirable, in that a single archive tape mount is required instead of individual mounts for each data set.

A deferred request is made by adding the DEFER parameter, described on page 192, to the ARCHIVE command.

Deferred archive requests are queued and kept in the ARCHCMDS subfile of the SAMS:Disk files data set. Once a queued request has been processed, its completion status is maintained for a default period of five days, after which time it is deleted automatically during the next DISPOSE run. The output from the DISPOSE run must be inspected to determine the outcome of processing for the request. Sysparm *DARCKEEP*, described on page 127 of the *Systems Guide*, may be used to alter the default period for which the status is maintained in the ARCHCMDS file.

Member Level Archive

A subset of a partitioned data set may be designated by specifying a list of one or more of its members or member alias names. Associated member and alias names are always maintained, regardless of which are specified in the list.

A member level archive is requested by adding the MEMBER= parameter, described on page 193, to the ARCHIVE command.

Restrictions

Deferred Archive

A subset of a PDS (specified by the MEMBER= parameter) is not supported.

Syntax

The syntax of the ARCHIVE command is as follows:

```
ARCHIVE DISP=,RETPD=,EXPDT=,ACCESS=,RPT=CIA,CIAOPT=
        DEFER,NEWNAME=,NEWHLQ=,MEMBER=
```

DISP=

The disposition for each data set successfully processed. The default value is DELETE.

Specify RECAT for recatalog to the SAMS:Disk pseudo-volume following deletion. This will make the data set available for auto-restore by the SAMS:Disk Catalog Management hook. The data sets can later be made unavailable for auto-restore by the SAMS:Disk Catalog Management hook by the MERGE command UN-CATALOG parameter.

Specify NOCAT to have SAMS:Disk take no catalog action against the data set. This will cause SAMS:Disk to delete the data set from DASD, and to leave it cataloged to its original volser.

Specify KEEP if the data set is not to be deleted, nor have its catalog status changed. Even better than specifying DISP=KEEP is to use the BACKUP command.

RETPD=

By default, data sets are assigned a retention period equal to the value of sysparm RETRETPD. To override the default, specify a one- to five-digit number indicating the number of days the selected data sets are to be retained in the SAMS:Disk archives. For related information, review the sysparm description of EXPDTUSE on page 143 in the *Systems Guide*.

If multiple ARCHIVE (and BACKUP) commands are entered, each may have a different RETPD= or EXPDT= value.

You may not specify both the RETPD and EXPDT parameters.

EXPDT=

By default, data sets are assigned a retention period equal to the value of sysparm RETRETPD. To override the default, specify a date constant in any acceptable SAMS:Disk format. For a list of valid dates, turn to the topic *"Date Formats"* on page 8. For related information, review the sysparm description of EXPDTUSE on page 143 in the *Systems Guide*.

SAMS:Disk treats DSNINDEX Julian expiration dates "99000" (except as described in the next paragraph), "99365", "99366" and "99999" as never scratch dates. That is, data sets whose ARCHVOLS or DSNINDEX records use these dates as expiration dates will not be deleted automatically as expired by the SAMS:Disk function IXMAINT.

By default, SAMS:Disk treats Julian expiration date "99000" as a never scratch date. But if you set sysparm UNCATDEL to a value of Y, SAMS:Disk treats Julian expiration date "99000" as a catalog control date. That is, data sets backed up or archived with an expiration date of 99000 will be deleted automatically by the SAMS:Disk function IXMAINT when the data sets become uncataloged. What this means is, if your catalog goes bad or someone uncatalogs some data sets, all back-ups and archives of these data sets will be deleted on the next IXMAINT run. Although you may use EXPDT=99000 for any purpose, it was intended for use with archiving generation data groups (GDGs), the generations of which are managed by the catalog. For example, to archive all GDG generations, leaving the two most recent on disk, deleting generations that roll off of the catalog, specify sysparm UNCATDEL with a value of Y and:

```
SCAN REALVOLS
SELECT CRITERIA=(GDGGEN,LE,-2)
ARCHIVE DISP=RECAT,EXPDT=99000
```

If multiple ARCHIVE (and BACKUP) commands are entered, each may have a different RETPD= or EXPDT= value.

You may not specify both the RETPD and EXPDT parameters.

ACCESS=

You may override the value of sysparm VSACCESS for this BACKUP command by specifying ACCESS= with a value of C for cluster-level access, or L for logical-level access.

RPT=CIA

This option allows the Cluster Internal Analysis (CIA) report to be produced as a by-product of live mode archive of VSAM clusters. The CIA report will not be produced during simulate runs. Refer to the description of this report on page 103 for further information.

CIAOPT=

If the RPT=CIA parameter is specified, this parameter can be used to specify which options are to be used. Refer to the description of this option on page 103 for further information.

NEWNAME=

An alternative data set name to be assigned to the DSNINDEX record for the archived data set (a relative GDG name is also permitted). The archived version is indexed under this name, which may be up to 44 characters in length.

DEFER

This parameter may be specified on the ARCHIVE command to indicate that the request is to be queued for later processing. For an overview of this function, turn to the topic *"Deferred Request"* on page 190.

NEWHLQ=

An alternative high level qualifier can be assigned to each data set selected for archive (a relative GDG name is also permitted). The archived version is indexed under this name, which may be up to 44 characters in length.

MEMBER=

To archive a subset of a partitioned data set, this parameter is used to specify one or more member and/or pattern names to constitute the archived version of the data set. A valid pattern for member level processing is any character followed by the '/' pattern character. This is the only valid pattern character available for this parameter. As an example:

```
MEMBER=(A/,GEN/)
```

For detailed information on the '/' pattern character, please turn to its description on page 6.

A maximum of 25 entries is supported and is only valid when an immediate archive is being done (not a deferred archive).

For an overview of this function, turn to the topic *"Member Level Archive"* on page 190.

ARCHIVE Command Examples

Example 1

Archive the unneeded data set LABS.EDP SOURCE for 60 days.

```
//jobname JOB (acct,info),etc.
//STEP EXEC DMS
FIND DSN=LABS.EDP SOURCE
ARCHIVE RETPD=60
```

Example 2

Archive for 7 years, allowing auto-restore, non-VSAM data sets and VSAM clusters not used in 28 days, except for SYS data sets and XDM/RD database components.

```
//jobname JOB (acct,info),etc.
//STEP EXEC DMS
PASSWORD MASTER=the-SAMS:Disk-master-password
SCAN REALVOLS
SELECT CRI=(USED,LE,TODAY-28) not used in 28 days
EXCLUDE DSNAMES=SYS/ leave my SYS/ on disk
ARCHIVE DISP=RECAT,RETPD=2557
```

Example 3

Archive, allowing auto-restore, generation data group data sets, except leave the two most recent generations on disk. Have these data sets under "catalog control",

which means that they are to be expired from the SAMS:Disk archives when they roll off of the catalog (see the JSCVSUT DEFINE GENERATIONDATAGROUP command LIMIT parameter).

```
//jobname JOB (acct,info),etc.
//STEP EXEC DMS
SCAN REALVOLS
SELECT CRITERIA=(GDG,EQ,YES,AND,GDGEN,LE,-2)
ARCHIVE DISP=RECAT,EXPDT=99000 catalog control
```

Example 4

Archive for 7 years, allowing auto-restore, non-VSAM data sets and VSAM clusters not used in 28 days, except for SYS data sets.

Also archive, allowing auto-restore, generation data group data sets, except leave the two most recent generations of most GDGs on disk, and leave the five most recent generations of the rest. Have these data sets under “catalog control”, which means that they are to be expired from the SAMS:Disk archives when they roll off of the catalog (see the JSCVSUT DEFINE GENERATIONDATAGROUP command LIMIT parameter).

```
//jobname JOB (acct,info),etc.
// EXEC DMS
PASSWORD MASTER=the-DMS-master-password
SCAN REALVOLS
EXCLUDE DSNAMES=(data.set.name.patterns.that.should,
                  have.more.than.two.generations.left.on.disk),
SELECT CRITERIA=(GDG,EQ,YES,AND,GDGEN,LE,-2)
ARCHIVE DISP=RECAT,EXPDT=99000
SELECT CRITERIA=(GDG,EQ,YES,AND,GDGEN,LE,-5)
ARCHIVE DISP=RECAT,EXPDT=99000
SELECT CRI=(USEDT,LE,TODAY-28)
EXCLUDE DSNAMES=SYS/
ARCHIVE DISP=RECAT,RETPD=2557
```

DELETE Command

Note: This command should be used only with the utmost care. Extreme — and unrecoverable — DATA LOSS can occur if used improperly.

The DELETE command instructs SAMS:Disk to delete the non-VSAM data sets and VSAM clusters that satisfy the selection criteria WITHOUT taking a backup copy of the data set.

Use this command with the utmost care. Always run it in SIMULATE mode first to determine exactly which data sets will be deleted. Keep in mind that because SAMS:Disk does not make a backup copy of the data set during delete, its recoverability (if required) may be in serious jeopardy. Thus, by default, this command is not allowed by SAMS:Disk. To use this command you must have specified sysparm DELETEOK with a value of Y.

Even with DELETEOK, this function will not allow the deletion of an DMF/EF Type-2 or VSAM catalogs. This function will also not allow the deletion of data sets beginning with “SYS1” unless you have sysparm SCRASYS1 with a value of Y.

DELETE DISP=

DISP=

This is the catalog action for each data set successfully processed. UNCAT is the default value. This value causes SAMS:Disk to uncatalog the data set when it is deleted.

Specify DISP=RECAT to recatalog to the SAMS:Disk psuedo-volser after deletion. This makes the data set a candidate for auto-restore through the SAMS:Disk Catalog Management hook. For the auto-restore to be successful, the data set must have been previously backed up. **CAUTION:** This disposition parameter in conjunction with the system parameter RECATPSUr, will now catalog to the psuedo-volser an uncataloged version of the selected data set. The recommended setting, unless you specifically know you want the uncataloged version, would be RECATPSUn, which limits only cataloged versions to be selected for cataloging to the psuedo-volser.

Specify DISP=NOCAT to cause SAMS:Disk to take no catalog action against the data set. This causes SAMS:Disk to delete the data set from DASD and leave it cataloged to the original volser.

DELETE Command Examples

Example 1

DELETE (in simulate mode) the unneeded cataloged data set LABS.EDP.SOURCE.

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
SET MODE=SIMULATE  
FIND DSN=LABS.EDP.SOURCE  
DELETE
```

Example 2

Delete (in simulate mode) system temporary data sets and any incorrectly cataloged non-VSAM data sets that are at least two days old.

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DMS  
SET MODE=SIMULATE  
SCAN REALVOLS  
SELECT CRITERIA=(TEMP,EQ,YES,AND,CREDIT,LE,TODAY-2)  
SELECT CRITERIA=(CATSTAT,NE,YES,AND,CREDIT,LE,TODAY-2)  
DELETE
```

EXPIRE Command

The EXPIRE command instructs SAMS:Disk to expire non-VSAM data sets and VSAM clusters. Non-VSAM data sets are expired based on the format-1 EXPDT. The catalog entry expiration date is used for VSAM clusters. The data set expiration date is compared against the current date, or the date specified in the DATE= parameter, to determine if a data set is expired. If a data set satisfies DSCL selection and is determined to be expired by SAMS:Disk, it is deleted from DASD.

EXPIRE DATE=

DATE=

This optional parameter specifies the date upon which to base expire processing. Specify a date constant in any acceptable SAMS:Disk date format. For a list of valid dates, turn to the topic *"Date Formats"* on page 8. The date specified in this parameter is compared against the data set expiration date. This comparison is used to determine if a data set should be expired. This parameter is useful in expiring data sets with expiration dates that have not yet been reached, by specifying a date greater than or equal to the not yet reached data set expiration date.

EXPIRE Command Examples

Example 1

Expire (in simulate mode) the expired data set LABS.KSV.SOURCE.

```
//jobname JOB (acct,info),etc.
//STEP EXEC DMS
SET MODE=SIMULATE
FIND DSN=LABS.KSV.SOURCE
EXPIRE
```

Example 2

Expire (in simulate mode) all data sets whose expiration dates are less than today's date and have not been used in one year, which reside on the DMSK01 volume.

```
//jobname JOB (acct,info),etc.
//STEP EXEC DMS
SET MODE=SIMULATE
SCAN REALVOLS
SELECT VOL=DMSK01,CRITERIA=(EXPDT,LT,TODAY,AND,
USEDT,LE,TODAY-365
EXPIRE
```

Example 3

Expire (in simulate mode) all VSAM clusters with data set names beginning with LABS.KSV., having expiration dates of January 1, 1999 or before.

```
//jobname JOB (acct,info),etc.
//STEP EXEC DMS
SET MODE=SIMULATE
SCAN CAT=/
SELECT DSN=LABS.KSV.,CRITERIA=(DSORG,EQ,VSAM)
EXPIRE DATE=1999.001
```

Sample Archive/Backup/Delete Report

1989.052	FEB 21, 1989	B A C K U P / A R C H I V E / D E L E T E P R O C E S S I N G												PAGE	1
TUESDAY	4.25 PM													SAMS:Disk	8.2
DATA SET NAME / NEWNAME		VOLUME	DEVTYPE	DSORG	RECFM	BLKSZ	LRECL	OPTCD	KEYL	ARC	SCR	CAT	REASON	KBYTES	TRACKS

LABS.EDP.CNTL		LABS82	3380	PO	FB	4,080	80	00	0	A	N	N	BACKUP	8,545	180
LABS.EDP.FILES		LABS83	3380	DA	F	23,476	0	01	0	A	N	N	BACKUP	1,424	30
LABS.EDP.LOADLIB		LABS82	3380	PO	U	6,144	0	00	0	A	N	N	BACKUP	569	12
LABS.EDP.MULTIVOL		LABS50	3350	PS	U	256	0	00	0	A	S	U	ARCHIV	2,822	148
		MVS001													
		LABS53													
		LABS54													
		LABS56													
		WRK002													
		SYSCAT													
		CAT001													
		LABS51													
		WRK001													
		ADS050													
LABS.EDP.MULTIVS		WRK800	3380	AM	KSDS	16,384	1,600	00	4	N	S	U	DELETE	17,803	375
LABS.EDP.PMRFILE		WRK800	3380	AM	KSDS	16,384	1,600	00	4	A	N	N	BACKUP	0	0
LABS.EDP.SOURCE		LABS53	3350	PO	FB	4,080	80	00	0	A	S	U	ARCHIV	8,466	444
LABS.EDP.SYSTERM		LABS52	3350	PS	FMB	4,840	121	00	0	N	S	U	DELETE	19	1
LABS.KSV.SOURCE.OLD		LABS51	3350	PO	FB	4,080	80	00	0	N	S	U	EXPIRE	8,466	444
		TOTAL		TOTAL		TOTAL									
		DATASETS		TRACKS		KILOBYTES									
		-----		-----		-----									
ARCHIVED/BACKED UP...		6		814		21,826									
SCRATCHED.....		5		1,412		37,576									
* AN ASTERISK BEFORE DATA SET NAME INDICATES AN ERROR OCCURRED WHILE PR OCESSING THE DATA SET															

Figure 5-2. Sample BACKUP/ARCHIVE/DELETE Report

EXPLICIT ARCHIVE

General Information

The SAMS:Disk archive function allows the user to archive (and subsequently to restore) data sets of any of the following organization types:

PS	Physical Sequential
DA	Direct Access
IS	Indexed Sequential
PO	Partitioned Organization
AM	VSAM

Device Independence and Ease of Use

Data sets of all organization types can be relocated to both new device types and new track locations. This includes VSAM data sets, traditionally regarded as unmovable. All that is needed is the data set name and volume. Data set attributes, space requirements, device conversion calculations, and location of data sets on the archive volumes are all maintained by SAMS:Disk. This permits a single fixed set of JCL to be used for all archives (and restores), without need of looking up any of the previously mentioned items, or modifying any of the JCL (typically DD statements for needed volumes).

Data Set Compression and Reorganization

When a data set is being read from disk, SAMS:Disk processes the data blocks within it “sequentially” as dictated by the organization type. Thus only active and used data blocks are read. This results in PDS and VSAM data sets being copied to the archive tape in compressed format. Any subsequent restore or volume recovery results not only in the consolidation of several extents into one (defragmentation), but also in the compression/reorganization of the data within the single extent (eliminating the additional need to compress and reorganize these same data sets).

PDS Member Lists

A subset of a partitioned data set may be designated by specifying a list of one or more of its members or member alias names. Associated member and alias names are always maintained, regardless of which are specified in the list. It should be noted that standard default processing at successful conclusion of the archive is to uncatalog and delete the PDS, and to index the archived members under the PDS name. Overrides to the defaults may be desirable in such a case.

PDS Anomalies

SAMS:Disk does extensive testing for errors and inconsistencies in partitioned data sets, and provides installation options for processing them even when logical errors

have been detected. Standard default processing is to issue diagnostic messages for the errors and then bypass them. Consult with your local SAMS:Disk installation and support staff to determine what processing options are in effect if any of your partitioned data sets are diagnosed to have anomalies.

Deferred Archives (Queued Archive Requests)

Requests for data set archival may be deferred over a period of time until a batch DISPOSE run can be performed to process all requests deferred since the previous batch run. Use of this facility is desirable, in that a single archive tape mount is required instead of individual mounts for each data set.

A deferred request is made by using the DARCHIVE command name, instead of ARCHIVE, or by adding the DEFER parameter to the ARCHIVE command.

Deferred archive requests are queued and kept in the ARCHCMDS subfile of the SAMS:Disk files data set. Once a queued request has been processed, its completion status is maintained for a default period of five days, after which time it is deleted automatically during the next DISPOSE run. The output from the DISPOSE run must be inspected to determine the outcome of processing for the request. Sysparm DARCKEEP may be used to alter the default period for which the status is maintained in the ARCHCMDS file.

Default Delete, Catalog, and Change Bit Resetting Actions

The ARCHIVE command defaults to deleting and uncataloging the named data set. The “change bit” in the VTOC entry for the data set is also turned off by default if the data set is not deleted. All three of these default actions can be changed by the use of sysparms EXPASCOP, ARCATOPT and EXPARSOP. Consult your local SAMS:Disk installation and support personnel to determine what defaults are being used.

Each default can be overridden by specifying the appropriate parameter on the ARCHIVE command.

Archive Report Sequencing Options

The archive report can be produced in three sequences:

1. data set name sequence
2. data set name sequence within volume
3. exact processing sequence (unsorted)

By default, all three reports are produced. See sysparm *ARCHSORT*, described on page 119 of the *Systems Guide*, if you want to change the defaults. Any combination of the three report sequences may be specified.

Explicit ARCHIVE Condition Codes

The following return codes are set at the completion of the ARCHIVE job step:

Table 5-1. Explicit ARCHIVE Condition Codes

Code	Description
0	All ARCHIVE/DARCHIVE requests have completed successfully.
1	All ARCHIVE/DARCHIVE requests have completed successfully. Produce sorted reports.
4	An error occurred during the archive of one or more data sets.
5	An error occurred during the archive of one or more data sets. Produce sorted reports.
16	One or more of the input commands contained an error and was not processed.
17	One or more of the input commands contained an error and was not processed. Produce sorted reports.

Restrictions

Unmovables

Data sets that are truly marked unmovable (not to be confused with VSAM, traditionally regarded as unmovable by many programs) are automatically bypassed unless sysparm UNMOVEAR is specified with a value of Y. SAMS:Disk Restore capabilities do not include absolute track allocation or track image processing, both of which are required to ensure the integrity of these special data sets. Therefore, if “unmovable” data sets are archived and/or restored, SAMS:Disk will not guarantee their integrity.

SYSCTLG Archival

An OS CVOL catalog may be archived under any name desired, but it must be given the name of SYSCTLG when it is restored. Connect pointers or alias entry pointers from the master catalog to the CVOL catalog are not altered. If a CVOL catalog is to be moved to a new volume, the user must ensure that the appropriate connects are done in order to use the moved catalog. Due to the embedded TTR pointers in a SYSCTLG, it must be restored to a volume of the same device type (all variations of the 8598 are considered the same type).

Deferred Archive

A subset of a PDS (specified by a member list) is not supported.

GDG Data Sets

Explicitly archiving GDG data sets is supported, but they must be specified in descending sequence.

Parameters

Requests to archive one or more non-VSAM data sets or VSAM clusters immediately or to have them placed into a queue to be processed at a later time (deferred requests) are submitted with the following JCL. The addition of a STEPCAT DD statement may be needed for processing incorrectly cataloged VSAM clusters.

```
//jobname JOB (acct,info),etc.
//STEP EXEC ARCHIVE
MODESET, DARCHIVE,
      AND/OR
ARCHIVE COMMANDS ENTERED HERE
MODESET SIMULATE
DARCHIVE
      SIMULATE,DSNAME=,VOLUME=,NEWNAME=,RETPD=,EXPDT=,
      RECATALOG,UNCATALOG,NOCATALOG,SCRATCH,NOSCRATCH,
      RESET,NORESET,PASSWORD= (PASSWORD= not applicable to VSAM)
ARCHIVE
      same parameters as above with the addition of
      MEMBER= (not applicable to VSAM),DEFER
```

Explicit ARCHIVE MODESET Command Parameters

The following is a description of the commands and parameters used to invoke the explicit archive function.

The MODESET command has been provided to allow explicit archival to be run in a simulated mode. If used, it must be the first command in the SYSIN input stream and will apply to all the following DARCHIVE or ARCHIVE commands.

```
MODESET SIMULATE
```

SIMULATE

Inclusion of this parameter will allow all explicit archive functions to be tested except for actual data set disposition and copy. The effect of the archive processing will be reflected in the SAMS:Disk report automatically generated.

No tape allocation is required in simulate mode.

EXPLICIT ARCHIVE and DARCHIVE Command Parameters

```
DARCHIVE
    SIMULATE,DSNAME=,VOLUME=,NEWNAME=,RETPD=,
    EXPDT=,RECATALOG,UNCATALOG,NOCATALOG,SCRATCH,
    NOSCRATCH,RESET,NORESET,PASSWORD=
ARCHIVE
    same parameters as above with the addition of
    MEMBER=,DEFER
```

SIMULATE

This simple parameter may be placed on each DARCHIVE or ARCHIVE command in lieu of using the MODESET SIMULATE command. For a given input stream, simulate and non-simulate modes should not be intermixed.

DSNAME=

The full explicit name of the non-VSAM data set or VSAM cluster to be archived (relative GDG version number is permitted, but if multiple generations are processed in the same job step, they must be specified in descending order). The catalog is searched to find VSAM clusters. The catalog is also searched to find non-VSAM data sets unless you also supply the volume parameter.

VOLUME=

This optional parameter specifies the volume serial number of the device containing the data set to be archived. If the data set is not found on this device, the request is rejected.

NEWNAME=

An alternative data set name to be assigned to the archived data set (a relative GDG name is also permitted). The archived version is indexed under this name, which may be up to 44 characters in length.

RETPD=

A one- to five-digit number to specify a retention period to be assigned to data sets. If this parameter is omitted, a default value is obtained from sysparm ARCRETPD.

EXPDT=

An expiration date in an accepted SAMS:Disk format may alternatively be specified to designate the period of time the data set will remain archived. (If the value given is less than the current date, the current date will be used as the EXPDT.)

MEMBER=

To archive a subset of a partitioned data set, this parameter is used to specify one or more member and/or group names to constitute the archived version of the data set. (A maximum of 25 entries is supported.)

This parameter is valid only when an immediate archive is being done (not a deferred archive).

Catalog Actions

Only one of the following three catalog parameters should be specified on any given command. However, if multiple parameters are entered,

- RECATALOG supersedes UNCATALOG
- UNCATALOG supersedes NOCATALOG

RECATALOG

The presence of this parameter causes the data set to be recataloged to the pseudo-volume after it has been archived.

UNCATALOG

The default value is UNCATALOG. The presence of this parameter causes non-VSAM data sets to be uncataloged after they have been archived successfully.

For VSAM clusters, this UNCATALOG parameter has no effect, use the SCRATCH parameter instead.

NOCATALOG

The presence of this parameter causes non-VSAM data sets to be left cataloged after they have been archived successfully. If you specify this NOCATALOG parameter, you probably want to specify NOSCRATCH as well.

For VSAM clusters, this NOCATALOG parameter has no effect, use the NOSCRATCH parameter instead.

Delete Actions

Only one of the scratch parameters should be specified on any given command. However, if both are entered,

- SCRATCH supersedes NOSCRATCH

SCRATCH

The default value is SCRATCH. This parameter may be specified to cause the data set to be deleted (and for VSAM, also uncataloged) after it is archived successfully.

NOSCRATCH

This parameter may be specified to prevent the deleting (and for VSAM, any catalog action) of the data set after it is archived successfully.

Reset Change Bit Actions

Only one of the reset parameters should be specified on any given command. However, if both are entered,

- RESET supersedes NORESET

RESET

The default value is RESET. This parameter will cause the “change bit” to be reset (to zero, or off) for any data set that is archived successfully.

NORESET

If a data set is backed up, normal processing will reset (to zero) the change bit DS1IND02 in the format-1 DSCB. Specification of this parameter will cause processing to bypass updating the change bit.

DEFER

This parameter may be specified on the ARCHIVE command to indicate that the request is to be queued for later processing. It may be used in lieu of changing the ARCHIVE command to DARCHIVE.

PASSWORD=

The password parameter provides the password value required to access a protected VSAM cluster. If the cluster is to be archived but not deleted, the read-level or higher access is required. If the cluster is to be deleted, the master-level password is required.

Processing Queued ARCHIVE Requests

To dispose of the deferred archive requests (that is, actually do the archiving) the JCL for immediate archival is used, with the exception that the module executed is ADSDM276. The following JCL will process (dispose of) all deferred archive requests placed in the ARCHCMDS queue:

```
//jobname JOB (acct,info),etc.
//STEP EXEC ARCHIVE,PARM.ARCHIVE=ADSDM276
```

Listing of Deferred Archive Request (LISTREQ)

The status of the deferred archive requests may be listed using the following JCL:

```
//jobname JOB (acct,info),etc.
//STEP EXEC LISTREQ
LISTREQ DSN=,ARCHIVE,ALL
```

DSNAME=

A list of one or more data set names or prefixes for which request entries are to be listed. If this parameter is not specified, the ALL parameter as defined below must be specified. (DSN=/ is equivalent to the ALL parameter.)

ARCHIVE

The presence of this parameter results in queued DARCHIVE request records being selected. It may be abbreviated as ARC.

This parameter **MUST** be specified to list deferred archive requests. The default value for LISTREQ will display deferred restore requests.

ALL

The presence of this parameter results in all the DARCHIVE request records being selected.

Erasing of Deferred ARCHIVE Requests

The DERASE command may be used with the following JCL to delete requests from the DARCHIVE queue:

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DERASE  
DERASE DSNAME=,ARCHIVE
```

DSNAME=

A list of one or more data set names or prefixes for which deferred requests are to be deleted. An acceptable abbreviation for this keyword is DSN. A maximum of 20 data set names or prefixes are allowed.

ARCHIVE

The presence of this parameter indicates that the deferred requests to be deleted are archive transactions. It may be abbreviated ARC.

Chapter 6. RESTORE/RECOVER

The SAMS:Disk restore function allows you restore DA, PO, POE and PS data sets and OS catalogs without the need for preallocation. You may also restore VSAM clusters and DMF/EF Type-2 catalogs. In most cases, you can even restore to a device type other than that from which the data set was backed up or archived.

- Physical Sequential (PS)
- Direct Access (DA)
- Indexed Sequential (IS)
- Partitioned Organization (PO)
- Access Method—VSAM (AM)

An individual data set or cluster is restored via the RESTORE command. A group of data sets and clusters, an entire volume, or a group of volumes, however, are restored via the RECOVER command.

Commands Available:	JCL Proc Executed:
DERASE	DERASE
LISTREQ	LISTREQ
RECOVER	FMS
RESTORE	RESTORE
(dispose)	RESTORE with overridden EXEC parm

General Information

The restore function begins by reading in all of the restore commands and sequencing them by file number within each archive tape (or archive data set on disk), the optimum performance arrangement. All data sets being restored from the archives on disk are done before those archived to tape. Two different variations of the restore function are available. The first form instructs SAMS:Disk to go immediately to the index of archived data sets, find the named data set and proceed with the restore. The second form instructs SAMS:Disk to begin by doing a catalog locate for the named data set. If the data set is cataloged to the SAMS:Disk pseudo-volume (default is ARCIVE), the same process as for the first form is continued. Otherwise a return code is set and the next restore request is processed. This second form of the RESTORE command is indicated by coding the SYSCTLG parameter on the command statement. The forms may be intermixed in the request stream.

Multiple Files Data Set for Restore

Multiple files data sets are supported for Restore, Deferred Restore, and DSCL Recover processing. If your installation maintains multiple files data sets and you want to make them all available to these functions, you should use the MFILES DD statement instead of the FILES DD statement. Add each files data set name to the MFILES concatenation, up to 256 files data sets are supported. The order of the concatenation determines the order of search for the data set. The first files data set that contains an index record for the archived data set will be used, even if another files data set contains a more recent version of the data set. Therefore, it is very important that files data sets be specified in the proper order.

Note: The FILES DD statement will be ignored if there is also an MFILES DD statement in the procedure. Concatenated files data sets are not supported with the FILES DD statement. Finally, the MFILES DD statement is not supported in dispose processing. If you have more than one files data set, you must execute DISPOSE separately for each one.

Restore Condition Codes

The following return codes are set at the completion of the restore job step:

Table 6-1. Restore Condition Codes

Code	Meaning
0	All data sets requested to be restored were restored successfully
2	Disposition processing (requested/default catalog actions) failed for one or more data sets
16	One or more Attempted restores failed

Non-VSAM Restore Volume Selection

The philosophy behind the restore allocation technique is to maximize system catalog integrity and reduce the creation of disconnected data sets. Therefore, the sequence for selecting a target volume for the restored data set is as follows:

1. VOL= parameter specified.

Allocation uses the specified volume. If the allocation fails to the specified volume, the restore is terminated. (mutually exclusive of POOL parameter).

2. POOL= parameter specified.

A pool name from the specified parmlib member DASDPOOL is used. If allocation fails for all volumes within the specified pool, the restore is terminated. (mutually exclusive of VOL parameter).

3. No directional parameters specified. (VOL= parameter or POOL= parameter omitted).

Allocation is directed to the original source volume. If allocation to the original source volume fails, an allocation attempt is made to the default pool. If a default pool is not specified and the allocation fails, the restore is terminated.

Note: Please review sysparms *RESCRCAT*, *RESCHCAT*, and *RECCHCAT*, beginning on page 158 of the *Systems Guide*. Under certain conditions, the allocation results may vary.

Multi-volume Restore

A multi-volume data set can be restored three different ways:

1. Sysparm PRIALLOC set to 'Y'.

The multi-volume data set will be allocated to the original volumes from which it was archived. If allocation fails, the restore is terminated.

Note: SAMS:Disk pooling is bypassed for multi-volume data sets when PRIALLOC is set to Y.

2. Sysparm PRIALLOC set to 'N'.

This setting consolidates the original total allocation to the new primary allocation amount. The multi-volume data set allocation is directed to a single volume, and if it does not fit on the first volume, the other volumes will be used as “overflow” volumes. If allocation fails, the restore is terminated. For example, if a 3-volume data set is restored

to a single volume, the first volume will contain a format-1 entry and a catalog entry with the 2 remaining volumes represented only by the catalog entry. (All volumes must be online)

3. Sysparm PRIALLOC set to 'N' and VOL= or POOL= parameter specified.

This setting directs the allocation to a volume using the consolidated primary allocation amount. If allocation fails, the restore is terminated. For example, if a 3-volume data set is restored to the specified volume, that volume will contain both a format-1 and a catalog entry.

Deferred Restores (Queued Restore Requests)

Requests for restoring of data sets may be deferred over a period of time until a batch DISPOSE run is made. Use of this facility is desirable in that the number of tape mounts will be minimized. This is a consequence of the sequencing of perhaps several restore requests by archive tape volume serial and file number, and restoring all within a single mount and scan of the needed tapes.

A deferred request is made by adding the DEFER parameter to the RESTORE command (or using DRESTORE instead of RESTORE as the command name).

Deferred restore requests are queued and kept in the RESTCMDS subfile of the SAMS:Disk files data set. Once a queued request has been processed, its completion status is maintained for a default period of five days, after which it is automatically deleted during the next “dispose run”. The output from the dispose run must be inspected to determine the outcome of processing for the request. Sysparm DRESKEEP may be used to alter the default period for which the status is maintained in the RESTCMDS file.

Restore of SYSCTLGS

Any time an OS CVOL catalog is restored, it must be given the name SYSCTLG in order to do proper formatting. Current support also requires that SYSCTLG be restored to the same device type from which it was archived. Connect or alias entry pointers from the master catalog are not altered. If a SYSCTLG is restored to a different volume, the user must ensure that the appropriate disconnects and connects are done to enable its use.

Restore Grace Periods

An installation option exists that assigns a grace period to any restored data set, during which time it is exempt from processing by archive (new backup copies are still taken though). Note, however, that the grace period applies only to the volume where the data set is restored. A data set with the same name on a different volume will not be exempt. Similarly, moving a data set after it is restored will nullify its grace period.

For details, refer to the sysparm descriptions for *RESRETPD* and *RETEXCLD* beginning on page 161 of the *Systems Guide*.

Restore to Preallocated Data Sets

By default, SAMS:Disk will not write over a preallocated data set. This can be changed, however, via sysparms *PREALLOC* and *VSPREDEF*. The *SCRATCH* parameter may also be specified on the *RESTORE* command, which will cause a preallocated data set to be scratched and then reallocated (providing the *DSORG* of the target data set matches the *DSORG* of the data set being restored, otherwise the restore will fail). Sysparm *RESCRCAT* is for non-VSAM data sets only. It is used to determine when and from which volume SAMS:Disk will scratch the preallocated data set. If the sysparm *RESCRCAT* is set to *Y*, and the *SCRATCH* parameter is specified, SAMS:Disk will scratch the preallocated data set from the cataloged volume, before attempting any allocation. If the sysparm *RESCRCAT* is set to *N* and the *SCRATCH* parameter is specified, SAMS:Disk will only scratch the data set if it resides on the same volume to which SAMS:Disk has attempted its allocation. For more information, review the sysparm description for *RESCRCAT* on page 159 of *Systems Guide*.

If preallocated data sets are allowed, *SCRATCH* is not specified, and a partitioned data set is being restored, the *ERASE* parameter permits an existing PDS to be overlaid by the restored copy. If *ERASE* is not specified, the members being restored are merged into the PDS at the end of its current member data. However, if some of the members being restored already exist in the PDS, they will not replace the duplicate members unless the *REPLACE* parameter is specified.

PDS Member Restore

A subset of a PDS may be restored by specifying a member list. These members can be restored directly into an existing PDS as described above, or into a new data set of their own.

Any alias for a given member may be specified for selective restoring. The alias name will be used to restore the true member, then all alias names associated with it will be updated.

Restore Space Allocation

When a data set is archived, the actual space used is calculated and recorded. At restore time the space allocated is either the original size of the data set or the used space quantity, whichever is larger. For example, a restore will not reduce the primary space quantity, but will adjust it up for data sets that have gone into extents. The primary quantity is adjusted appropriately for restoring to different device types.

Restore Considerations for GDG Data Sets

When restoring GDG data sets, the following principles will be followed in regards to catalog action:

NEWNAME:

When this parameter is specified, RESTORE treats the target DSNNAME as a new data set creation. The target data set will always be cataloged and placed in the sphere, whether the sphere is full or not. The following occurs:

1. If a catalog entry exists, the entry is left intact
2. If no catalog entry exists, the data set will be cataloged unless doing so would force another entry out of the catalog (i.e., the sphere is full)

If SAMS:Disk can not set a data set's catalog entry to the status SAMS:Disk feels is correct, message 3729 will be issued to indicate the status the data set was left in, and the reason for placing the data set in that status.

For related information, refer to the sysparm description of *GDGCATEX*, located on page 144 in the *Systems Guide*.

Restore and Recover of VSAM Clusters

VSAM clusters are reorganized as they are restored or recovered (unless they were archived in control interval image copy format). Often, this eliminates multiple extents and CA/CI splits for the clusters involved. Catalog entries are dynamically recreated for each cluster with the attributes as they existed when the archive/backup was created (unless the cluster was predefined). SAMS:Disk adjusts space allocations for the device type to which the clusters are recovered.

When recovering suballocated (non-unique) VSAM clusters to a “new” volume, you must predefine the VSAM space. Once this space is created, SAMS:Disk dynamically defines the VSAM clusters into it. SAMS:Disk automatically defines the self-containing spaces for unique clusters. When restoring suballocated clusters that came from a VSAM catalog into an DMF/EF Type-2 catalog, the attribute is automatically changed to UNIQUE, regardless of its prior value.

The files data set contains the volume serial of the first data component of a VSAM cluster. During Archive/Backup processing, SAMS:Disk issues a message that shows this volume serial. The volume serial must be correctly specified for a RECOVER run that uses the VOL= parameter. SAMS:Disk can only RECOVER clusters if the first data component volume is correctly specified.

Restrictions in Restore and Recover of VSAM Clusters

Take care when altering the control interval size of any ESDS cluster. Changing the CI size can alter the way logical records fit in the data set when it is restored, causing internal RBA pointers to become invalid. If the cluster was archived in control interval image copy format, the CI size cannot be changed at restore time. Changing CI size causes a record length error when SAMS:Disk tries to restore the data into the data set.

Volume recovery can restore multi-volume data sets when the volume recovered is the primary volume for the cluster. Otherwise, the cluster is bypassed.

Recover does not process alternate indexes that are archived as physical data sets because of possible synchronization problems with the base cluster. SAMS:Disk redefines and rebuilds the cluster’s alternate indexes when the base cluster is restored.

Base clusters and physical alternate indexes cannot be restored in the same job step. The base cluster(s) must be restored first, followed by another job or step to restore the alternate indexes. For more information, please review the sysparm description for VSARCAIX on page 183 of the *Systems Guide*.

Special Processing Options

The following topics detail special processing options that may be used to tailor the VSAM restore and recover functions. Consult the *Systems Guide* for further explanation of available sysparms (page 95) and user exits (page 195).

Secondary Extent Consolidation (Defragmentation of VSAM Clusters)

For single volume data sets, SAMS:Disk will automatically change the primary allocation for a DEFINE request based on original allocated extents with sysparm VSSPUSED specified with a value of Y. By specifying this parameter, SAMS:Disk will sum all of the extents that were allocated to the data set at archival time. It will then determine the amount of this space that was actually used, and if this value is larger than the original primary amount, will use the new value as the primary allocation amount. For multi-volume data sets, SAMS:Disk will always use the original allocation amounts. If the data set is being restored to an unlike device type, the allocation amount will be changed to reflect either the increased or decreased track capacity.

In addition to sysparm VSSPUSED, the VSALLOEX user exit is available to allow a user program to change allocation amounts based on information supplied to the exit.

If a new allocation amount is specified on the RESTORE command via the ALLOC=, DALLOC= or IALLOC= parameters, the value specified will be used directly without any conversions.

Default Catalog Processing

Normal SAMS:Disk processing allows catalog management to determine which catalog to define the cluster in, based on the “Order of Catalog Use: Define” outlined in the Hitachi publication Access Method Services Reference. By specifying sysparm VSDEFCAT with a value of N, SAMS:Disk will define the cluster in the same catalog from which it was archived.

Note: This sysparm has no effect when the catalog is explicitly named on the RESTORE or RECOVER command.

Speed Option

When VSAM clusters are defined via Access Method Services, the normal default value of RECOVERY is used unless SPEED is explicitly requested. SAMS:Disk will automatically define all clusters with the SPEED option to improve restore performance, regardless of how they were originally defined. To use the option defined with the cluster, specify sysparm VSSPEEDO with a value of N.

Changing Data Sets Attributes

Either because of choice or necessity, there are times when the data set being restored needs to have some of its attributes modified. This can be particularly true when restoring to an unlike device type, where the cluster define may fail due to restrictions in buffer space and control interval size. By specifying sysparm VSATTRIB with a value of Y, SAMS:Disk will ignore the original buffer space and control interval size values that were defined with the cluster and will allow Access Method Services to calculate new values for these parameters.

In addition, the user may code an exit named VSATTREX. This exit not only allows the user to change the buffer space and control interval size, but it also allows the changing of certain attributes. The most notable of these allows the user exit to define the cluster as either UNIQUE or SUBALLOCATED, regardless of its original value. The attributes mentioned above, along with many others, may also be explicitly changed by specifying them on the restore command.

Empty Clusters

When restoring empty clusters, the only processing that takes place is the DEFINE. Therefore, the first time the data set is accessed after the restore, it must be processed in LOAD mode. A dummy record will be inserted and then deleted into any cluster that had a HURBA (High Used Relative Byte Address) greater than zero but had no data records in it. This option is useful primarily in an environment where the data set must be initially loaded in order to be used.

Deleting Unexpired Data Sets

By specifying sysparm VSMPURGE with a value of Y (the default value), SAMS:Disk will allow the deletion of unexpired data sets when the SCRATCH parameter is specified on a restore or recover. This is equivalent to using the PURGE parameter on an JSCVSUT delete request.

Last Use Date Maintenance

When a cluster is being restored, SAMS:Disk will check to see if it was initially defined in an DMF/EF Type-2 catalog (clusters in regular VSAM catalogs do not have the last use date maintained by open). If it was defined in an DMF/EF Type-2 catalog and the last use date was not binary zeros, SAMS:Disk will modify the last use date after the definition of the cluster takes place (the new definition must also be in an DMF/EF Type-2 catalog). The last use date will be changed to the current date if the CREATE parameter is specified on the RESTORE or RECOVER command. The cluster must have been archived with a release of DMS of 7.1 or higher for this support to be activated, regardless of your current level of SAMS:Disk.

Overriding Erase Option When Deleting

When a cluster is defined with the ERASE option, it means the data component's extents must be overwritten with binary zeros when it is deleted. Occasionally it is necessary to override this option in order to delete the cluster. When SAMS:Disk deletes a cluster, it uses the value with which the cluster was defined (ERASE or NOERASE). If sysparm VSNOERAS is specified with a value of Y and a cluster defined with the ERASE option fails deletion with a specific return code, SAMS:Disk will retry the deletion overriding the ERASE option (effectively making it a NOERASE option). This is equivalent to issuing an JSCVSUT DELETE command with the NOERASE parameter specified. Note that even with this sysparm specified, SAMS:Disk will attempt to delete the cluster first with its default value. Only if the sysparm is specified will SAMS:Disk retry the deletion. (The default value for this sysparm is N, which allows the deletion to fail and processing is bypassed for the cluster).

Using NOSCRATCH Option When Deleting

The normal mode of operation for a delete is to remove the cluster's definition from all components of the DMF/EF Type-2 catalog. Occasionally, however, it is necessary to delete the cluster entries from with the "noscratch" option prior to restoring the VSAM clusters. This can be done with JSCVSUT, or can automatically be invoked by SAMS:Disk by specifying sysparm VSNOSCRO with a value of Y.

With this parameter specified as a Y (the default value is N), SAMS:Disk will reattempt any delete that fails with a format-7 error (during a normal cluster recovery operation) with the noscratch option. Note that the noscratch option will only be attempted if the normal delete fails.

Processing Clusters in Use by Other Tasks

Normal SAMS:Disk processing determines the disposition type to use for data set allocation based on the share options with which a cluster is defined. If a cluster is defined with SHAREOPTIONS (1 3), it is allocated with DISP=SHR; otherwise it is allocated with DISP=OLD. By specifying sysparm VSALOSHR with a value of Y, all allocations will be made with DISP=SHR. In addition, user exit USERVENQ can be invoked to determine the allocation desired on the data set level.

Restoring to Predefined and Reusable Clusters

SAMS:Disk allows restoring to predefined and reusable clusters by specifying sysparm VSPREDEF with a value of Y. The following logic is employed by SAMS:Disk in determining when to use a predefined or reusable cluster's definition:

1. An attempt to define the entity is performed. If the define is successful, the cluster was not predefined and normal restore procedures are used.
2. If the define failed because the entity already exists, a check is made to determine if the SCRATCH parameter was specified on the RESTORE or RECOVER command. If it was specified, the cluster is scratched and redefined, regardless of the value specified in sysparm VSPREDEF.
3. If the define failed and SCRATCH was not specified, but VSPREDEF is specified with a Y, the cluster's attributes are retrieved from the current (predefined) definition. If it is defined as reusable, and is not an empty cluster, it is opened with the reusable option and the restore continues. Otherwise, the restore for the data set is bypassed.

Note: When restoring data sets using the predefined or reusable option, SAMS:Disk has to assume that the definition that has been found is indeed correct. Since it may not be, SAMS:Disk can assume no responsibility for the validity of the restored data. This option should only be used with the utmost care. Also note that empty clusters will not be restored over an existing data component when VSPREDEF is specified without the SCRATCH parameter.

Optional Cluster Data Report at Restore Time

By specifying sysparm VSRESCLD with a value of Y, SAMS:Disk will produce a Cluster Data Report (CLD) for each cluster successfully restored. The report lines are inserted after each cluster's normal one-line entry on the "restored data sets" report. The lines that are printed are determined by the values specified for sysparm RPTCLSEL. This report allows you to review the actual attributes of the restored cluster without requiring a separate report to be run.

Determination of Data and Index Component New Names

When clusters are restored using the NEWNAME parameter, SAMS:Disk must determine the new name to use for the data and index components. If the name was originally generated by AMS, SAMS:Disk will allow VSAM to generate the new name for the component. However, if the user specified the component name at define time, SAMS:Disk must generate a new name based on changes detected between the original base cluster name and the new name specified for the base cluster. In addition, the VSNEWNEX user exit is available to either specify a new component name or to instruct AMS to generate a new name. You may also assign new names on the RESTORE command via the DNAME and INAME parameters.

Determination of Alternate Index and Path New Names

When clusters with paths and/or alternate indexes associated with it are restored with the NEWNAME parameter, SAMS:Disk must generate new names for these components. The names that it will generate are based on a comparison of the original base cluster name and its new name, along with a comparison of the original base cluster name and the original component name. SAMS:Disk will be able to generate a much “cleaner” new name for its associated components if the new name that is specified for a cluster either:

1. Has only the first two node names changed, with the remainder of the new name the same:
 - a. Original name = LABS.TJP.PMRFILE
 - b. New base name = LABS.TJP2.PMRFILE
2. Has only a suffix added to the original base cluster name:
 - a. Original name = LABS.TJP.PMRFILE
 - b. New base name = LABS.TJP.PMRFILE.TESTFILE

If the new name generated by SAMS:Disk is unsatisfactory, you may assign your own new name. Refer to the *Examples* section on page 423 for details.

Restoring Alternate Indexes

Alternate indexes are automatically redefined and rebuilt (by invoking the BLDINDEX facility of JSCVSUT) when SAMS:Disk restores a base cluster that had alternate index associations when it was archived. You may bypass the alternate index define and/or rebuild in 1 of 3 ways:

1. By specifying NOAIXDEF and NOBLDINX on the Restore command
2. By specifying AIX=NO on the DSCL Recover command
3. By specifying AIX=DEFINEONLY on the DSCL Recover command

In addition, sysparms VSAIXDEF and VSBLDINX may be specified with a value of N to force SAMS:Disk to bypass alternate index processing as a default. For example:

Archive a base cluster and all of its alternate indexes:

```
FIND DSN=clustername
ARCHIVE DISP=RECAT
```

Restore a base cluster and rebuild all of its alternate indexes automatically:

```
RESTORE DSN=clustername
```

Restore a base cluster and rebuild two of its alternate indexes explicitly:

```
RESTORE DSN=clustername,NOAIXDEF
RESTORE DSN=clustername,AIXNAME=alternateindex1name
RESTORE DSN=clustername,AIXNAME=alternateindex2name
```

Recover under DSCL example:

```
RECOVER AIX=YES
```

Volume Selection for Alternate Indexes

Also, SAMS:Disk must decide the volumes to which the alternate indexes are to be restored. If the base cluster is being restored to its original volume, the alternate indexes will automatically be restored to their original volumes, even if they were different from the base's. If the base cluster is restored to new volume(s), SAMS:Disk uses the following logic for placement of the alternate indexes:

1. If the alternate index originally resided on the same volumes as the base cluster, the new volumes will be used for the alternate index as well.
2. the alternate index will go back to its original volume(s).

Restoring Alternate Indexes which were Archived Separately

In addition to the methods discussed above, alternate indexes can also be archived and restored separately from their base clusters. For example:

Archive an alternate index by itself (SAMS:Disk will retain the association information from the catalog in the files data set):

```
FIND DSN=alternateindexname
ARCHIVE DISP=RECAT
```

Restore an alternate index by itself and re-establish association with the base cluster:

```
RESTORE DSN=alternateindexname
```

Note: In order to restore an alternate index as discussed above, the associated base cluster must be preallocated and cataloged in the same catalog as the alternate index.

There are three critical precautions you must follow if you want to separately archive and restore alternate indexes:

1. In archive, since the base cluster is not being archived at the same time as the alternate index, it is imperative that no updates to either file occur after the first archival begins. SAMS:Disk cannot ensure this, so you must control this manually in order to keep the files synchronized. This becomes complicated if the data sets are selected implicitly by

DSCL, because SAMS:Disk processes the data sets in ascending data set name sequence without regard to the associations between the data sets (i.e. the base cluster/alternate index associations). Finally, if you are going to archive the alternate indexes separately, the best thing to do is explicitly archive the base cluster and all alternate indexes in a single archive job. For example:

```
FIND DSN=alternateindex1name
ARCHIVE DISP=RECAT
FIND DSN=alternateindex2name
ARCHIVE DISP=RECAT
FIND DSN=alternateindex3name
ARCHIVE DISP=RECAT
FIND DSN=clustername
ARCHIVE DISP=RECAT
```

2. In restore, since the base cluster is not being restored at the same time as the alternate index, it is imperative that no updates to either file occur until after the last restore completes. For example, if the base cluster is restored first and the alternate index second, if the base cluster is updated prior to the completion of the restore of the alternate index, the two files will be unsynchronized. This type of error can not be detected by SAMS:Disk and no error will be indicated. Once a base cluster and alternate index become unsynchronized, any application that accesses the base cluster through the alternate index may return erroneous information without the user being aware of any problem.
3. Another problem can develop if you attempt to separately restore an alternate index that was defined over an ESDS cluster. If you restore the base cluster with a different CISIZE than its defined original, your separately-restored alternate index will probably be invalid. This occurs because the alternate index contains pointers to the base cluster based on the relative byte address (RBA). When you restore the base cluster with a different CISIZE, the relative addresses of the records in the base cluster will probably change after the first control interval, thereby making all other offsets in the alternate index invalid. Although you can restore this alternate index, it cannot be used to access the data in the base cluster.

Volume Selection for Multi-volume Clusters

Please review “Volume Selection for Alternate Indexes” above.

The DSCL RECOVER command has a VOLUMES= parameter on the SELECT command that assist in identifying volumes from which data sets are to be recovered. If the first volume portion of any multi-volume cluster existed on one of the named volumes, all portions will be recovered to their appropriate volumes. If the TOVOL= parameter is specified, however, RECOVER interprets this to mean that all of the multi-volume portions are to be moved to the single volume specified,

which may not have space to contain it. If it does fit, the extra space that it used may prevent other clusters from being recovered.

The RESTORE command works similarly, but provides greater flexibility. If only the DSN= parameter is specified, all portions of the cluster will be restored to their appropriate volumes, whether it is a single or multi-volume cluster. A VOL= parameter is available that acts the same as the TOVOL= parameter described for RECOVER above; that is, all portions will be directed to the specified volume. However, parameters MVOL=(..list..), DVOL=(..list..) and IVOL=(..list..) are also available to supply multiple target volumes whenever a multi-volume cluster must be moved. The MVOL= list is sufficient if the data and index components may share the same volumes. The DVOL= and IVOL= lists are provided to allow these two components to be separated, and will override the MVOL= list if both are specified.

DMF/EF Type-2 Catalog Recovery

The volume recovery (not the restore) function supports the recovery of DMF/EF Type-2 catalogs. Although DMF/EF Type-2 catalogs are VSAM, the SAMS:Disk recover function regards them as neither VSAM nor non-VSAM, but special data sets with special recovery requirements. The CATALOGS= parameter on the SELECT command determines whether or not DMF/EF Type-2 catalogs are included in the recovery. When included in the recovery, SAMS:Disk always recovers the DMF/EF Type-2 catalogs first — before any of the data sets. This allows the catalogs to be updated correctly for those data sets that are subsequently included in the recovery process.

There are three parameters — CATALOGS=catname, DEFALIAS= YES/no, and EXPORTF=NO/yes — on the SELECT command that apply specifically to DMF/EF Type-2 catalog recovery. The CATALOGS= parameter provides a higher degree of control in deciding what catalogs are to be restored. It defaults to NO and must be specifically selected to be included. Please see “*DMF/EF Type-2 Catalog Backup and Recovery*” on page 176 for general information regarding the other parameters.

As explained in that section, SAMS:Disk dynamically links to Hitachi’s JSCVSUT IMPORT function to recover a catalog, and will then optionally redefine the alias entries in the master catalog. A few particulars regarding the IMPORT command should be noted. (See the Hitachi JSCVSUT manual for more specific information.) If the “old” catalog is still on the volume, it will be overlaid. A “new” catalog may be preallocated if desired (perhaps to increase the size), and the preallocated space will be used. If the catalog does not exist, IMPORT will define it. You cannot give the catalog a new name, however, as the IMPORT command does not support it. Another unexplained but observed phenomenon is that if the catalog was allocated in tracks, IMPORT will increase the allocation values if it has to redefine it. If track allocations were not used, IMPORT will reuse the original space values.

Files Data Set Updates during RESTORE/RECOVER

When a restore/recover occurs the files data set archive record for each data set is updated. This record is called the *DSNINDEX* record and is described on page 423 in the *Systems Guide*. Three fields are updated in the *DSNINDEX* record for each restore. The first two are the date and time of the restore. The third field is a count of the number of restores that have been done from that *DSNINDEX* record. The restore date is used for 3 functions:

1. The MERGE function uses this date in determining if a data set should be merged onto a different ARCHVOL
2. The IXMAINT function uses this date to determine if a *DSNINDEX* record should be removed from the files data set
3. The RECOVER function uses this date in conjunction with its RESDT criteria parameter

The restore time and count are currently for informational purposes only.

Use of Primary or Copy Archive Volumes

If a primary archive volume becomes unusable, the RESTORE/RECOVER functions can utilize the duplicate copy by disabling the primary volume. Refer to the *"Files Data Set IXMAINT"* chapter on page 279 for more information.

Default processing of the RESTORE/RECOVER functions will call for the primary archive volume (unless it has been disabled). If it is necessary to always use the duplicate volumes, such as in a disaster recovery situation, specify sysparm RESCOPYT with a value of Y.

Restrictions

The following is a list of restrictions that pertain to the restore/recover functions only. For other restrictions that apply to the SAMS:Disk product in general, review the *"General Restrictions"* topic on page 19.

- A SAMS:Disk Stand-Alone Restore (DMSSAR) function is provided as a separate product, not through the RESTORE or RECOVER procs. The Stand-Alone Restore function allows you to recover data sets critical to bringing up your operating system. You must produce a volume-level backup tape before you can use this function. (For further information, see the *VBACKUP/VRECOVER* section beginning on page 251 of this manual, and the *Stand—Alone Restore Guide*.)
- When recovering a volume using the VTOC parameter, only the most recent copy of the VTOC that exists in the archives will be used as a screening list.

RESTORE JCL

Use the following JCL to restore one or more data sets or clusters immediately, or to have them placed into a queue to be processed at a later time (deferred requests).

```
//jobname JOB (acct,info),etc.
//STEP EXEC RESTORE
        PASSWORD CATALOG=,MASTER=
        RESTORE  DSNAME=,NEWNAME=,VOLUME=,VERSION=,TIMEDATE=,SCRATCH,ERASE,CREATE,DEFER,
```

Figure 6-1. Sample RESTORE JCL

Plus the following parameter honored only when the DEFER parameter is not specified:

NOLOAD,

Plus the following parameters honored for non-VSAM data sets only:

POOL=,NOCATALOG,RECATALOG,SYSCTLG,

Plus the following parameters honored for non-VSAM data sets and only when the DEFER parameter is not specified:

MEMBER=,REPLACE,DIRBLKS=,

Plus the following parameters honored for VSAM clusters only:

PASSWORD=,

Plus the following parameters honored for VSAM clusters and only when the DEFER parameter is not specified:

CATALOG=,RELATE=,RELPSWD=,

- Alternate index parameters:

AIXNAME=,AIXDEF,NOAIXDEF,BLDINX,NOBLDINX,

- Cluster level parameters:

ALLOC=,FREESPACE=,RECSZ=,BUFFERSPACE=,CISZ=,MASTERPW=,
CONTROLPW=,UPDATEPW=,READPW=,ATTEMPTS=,OWNER=,
MVOLSERS=,FOR=,TO=,SHROPTIONS=,KEYS=,NOERASE,IMBED,
NOIMBED,ORDER,UNORDER,REPLICATE,NOREPLICATE,REUSE,
NOREUSE,SPAN,NOSPAN,SPEED,RECOVERY,UNIQUE,SUBALLOC,
WRITECHK,NOWRITECHK,

- Data component level parameters:

DNAME=,DALLOC=,DCISZ=,DMASTERPW=,DCONTROLPW=,
DUPDATEPW=,DREADPW=,DATTEMPTS=,DOWNER=,DSHROPTIONS=,
DVOLSERS=,DORDER,DUNORDER,DWRITECHK,DNOWRITECHK,

- Index component level parameters:

INAME=,IALLOC=,ICISZ=,IMASTERPW=,ICONTROLPW=,IUPDATEPW=,
IREADPW=,IATTEMPTS=,IOWNER=,ISHROPTIONS,IVOLSERS=,
IORDER,IUNORDER,IWRITECHK,INOWRITECHK

RESTORE Password Command

If you are restoring non-VSAM data sets only, or if you do not use password protection of your VSAM clusters, you may omit this PASSWORD command.

If the catalog in which VSAM clusters are to be defined is update-protected, the catalog's master password must be supplied via this PASSWORD command. In addition, if the cluster being restored was password-protected and its current password does not match that which is on the archive medium, this command must be supplied with the proper password so that the scratch of the existing data set will not fail.

```
PASSWORD CATALOG=,MASTER=
```

CATALOG=

Specify the catalog name to which the clusters are being restored.

MASTER=

Specify the catalog's master-level password.

RESTORE Command and Parameters

DSNAME=

Enter the full explicit name of the non-VSAM data set (relative GDG name is permitted), VSAM cluster or physical alternate index to be restored.

To restore alternate indexes based on the catalog information archived with a base cluster, enter the base cluster's name, not the alternate index names.

Note: Any time an OS CVOL catalog is restored, it must be given the name SYSCTLG in order to do proper formatting. Current support also requires that SYSCTLG be restored to the same device type from which it was archived.

NEWNAME=

The data set name to be assigned to the restored data set or cluster. A maximum of one entry is supported.

For GDGs, a relative GDG name may be specified.

For VSAM, this name applies only to the primary base cluster or alternate index being restored. If the data (and for a KSDS, the index also) component was originally assigned a name by Access Method Services, SAMS:Disk will let AMS generate a new name for the component(s) again (unless sysparms VSAPPEND, VSDATSUF, or VSINXSUF are specified).

If the component name(s) were originally specified by the user, SAMS:Disk will assign new names to the data and index components based on the differences detected between the original base cluster and the new name specified on the command. The data and index names may also be explicitly specified on the DNAME and INAME parameters.

If a base cluster is being restored (AIXNAME parameter is not specified) that had alternate index associations, the new name specified is the one to be given the base cluster. SAMS:Disk will generate names for its alternate index(es) and path(s).

If an alternate index is being redefined from a catalog definition found with the base cluster (AIXNAME parameter specified), the new name will be given to the alternate index being restored.

VOLUME=

The volume serial number of the device to which the data set is to be restored. The restore attempt will be terminated if the allocation can not be on the specified volume. This parameter is mutually exclusive with the POOL parameter, and a maximum of one entry is supported.

VERSION=

To restore other than the most recent version of the data set contained in the SAMS:Disk archives (either an archived copy or a backup copy), specify from -1 to -99 to get the successively older version of the data set.

TIMEDATE=

This parameter may be used to identify a specific version of the data set within the archives, rather than the relative version as supplied on the VERSION= parameter. Its use would be appropriate if a possibility exists that another version of the same data set will be placed in the archives before the restore can be completed.

The format for this parameter is “hhmmmyydd”. Correct values can be obtained by listing the SAMS:Disk index entries for the data set via either batch or TSS.

SCRATCH

Use this parameter to scratch a non-VSAM data set when it exists on the target volume (providing the DSORG of the target data set matches the DSORG of the data set being restored) with the same name that the data set being restored is to have. If sysparm RESCRCAT has a value of Y, SAMS:Disk will attempt to scratch the target data set from its cataloged volume prior to restoration. With sysparm RESCRCAT specified as N, SAMS:Disk will scratch the target data set only if it resides on the same volume to which SAMS:Disk attempted its allocation.

For VSAM, if a cluster already exists in the catalog with the same name that the cluster being restored is to have, SAMS:Disk will delete the target data set name, then reallocate the target data set name and restore the data.

ERASE

If a partitioned data set is being restored to a preallocated data set, this parameter causes the existing contents to be erased and overlaid with the restored version.

If ERASE is not specified when restoring to a preallocated PDS, sufficient space must be left to load the members following the existing data. (See the REPLACE parameter on page 227.)

If a VSAM cluster is being restored, this parameter indicates the cluster's data component is to be overwritten with binary zeros when its catalog entry is deleted.

CREATE

Specify this parameter if the creation date for the data set being restored is to be set equal to the current date. Default processing leaves the original creation date unaltered.

DEFER

This parameter may be specified on the RESTORE command to indicate that the request is to be queued for later processing. It may be used in lieu of changing the RESTORE command to DRESTORE.

Parameters Honored Without the Defer Parameter

NOLOAD

For non-VSAM data sets, this parameter causes SAMS:Disk to allocate the data set being restored, but not load any of its records.

For VSAM data sets, this parameter causes SAMS:Disk to define the cluster being restored, but not load any of its records. This can be useful if you want to restore a cluster's definition and then alter it before restoring data to it (as a preallocated cluster).

Note: This parameter cannot be used with DEFER.

Parameters for Non-VSAM Data Sets

POOL=

The name of the DASD pool to be used to find a volume on which to restore the data set. This is for non-VSAM data sets only (mutually exclusive with the VOL= parameter).

NOCATALOG

Suppress catalog action that would normally take place upon data set restoration.

RECATALOG

Default processing attempts to catalog all data sets, but will not change an existing catalog entry. Specify this parameter only when you want to force the data set to be cataloged to the target volume.

SYSCTLG

Code this parameter if the RESTORE request is to be conditioned on the named data set being cataloged to the pseudo-volume (implying that the data set does not exist on DASD but only in the SAMS:Disk archives).

Parameters for Non-VSAM Without DEFER

MEMBER=

A list of one or more member names or patterns to be restored from the archived data set may be specified. See parameters REPLACE, ERASE and SCRATCH also. A maximum of 25 entries is supported.

Note: This parameter cannot be used with DEFER.

REPLACE

This optional parameter permits members in an existing PDS to be replaced by duplicate named members being restored. Default processing merges in only those members that do not have conflicting names.

Note: This parameter cannot be used with DEFER.

DIRBLKS=(field1,field2)

This optional parameter permits the size of the directory to be adjusted. Field1 is a numeric value interpreted according to the value of field2. Field2 may be either PCT (for percent) or ABS (for absolute), and defaults to PCT if not specified.

- (nn,PCT) causes the directory to be adjusted such that, as close as possible, the new directory will be nn percent full.
- (nn,ABS) causes exactly nn directory blocks to be allocated, unless they are not enough to hold all of the entries. In this case, the minimum number that is sufficient will be used.

Note: This parameter is ignored when members are being merged into an existing PDS, and it cannot be used with the DEFER parameter.

Parameter for VSAM Clusters

PASSWORD=

This optional parameter is required only when attempting to restore a cluster that was archived with assigned passwords. The password specified must match the cluster's master password as found on the archive medium. Optionally, you may supply the SAMS:Disk master password.

Note: This parameter is required whenever a data set is being restored that had passwords assigned, regardless of whether the receiving catalog is protected or not.

Parameters for VSAM Without DEFER

Specify one or more of the following parameters only if you wish to change a particular value for the cluster from that which is found on the archive medium. If a value is given at both the cluster and component level (for example, ALLOC= and DALLOC=), the value specified at the data or index level will take precedence over the cluster value, just as JSCVSUT does when you issue a DEFINE. Refer to the Hitachi publication Access Method Services Reference for a more complete description of the following parameters.

CATALOG=

This optional parameter applies only to the restore of a VSAM cluster. It specifies the name of a VSAM catalog in which the restored cluster is to be defined (overriding the catalog selected by normal catalog management functions). If this parameter

is used, you should also specify the same catalog in a STEPCAT DD statement. If you don't, VSAM OPEN errors are likely to occur. This is caused by VSAM OPEN routines always following the standard catalog management rules to locate the proper catalog, regardless of the catalog specified for the define processing.

Note 1: For non-DMF/EF Type-2 catalogs, a named catalog must own the volume to which the cluster is being restored.

Note 2: Use caution when adding a STEPCAT, since all new catalog entries (non-VSAM as well as VSAM) will be placed in the STEPCAT catalog.

RELATE=

This parameter must be specified when you are restoring an alternate index (either a physical AIX or one from the catalog information archived with the base cluster) to a base cluster that was restored with a new name. Give the new name that the base cluster was restored with in this parameter.

RELPSWD=

This parameter must be specified when an alternate index is being restored and its base cluster was defined with passwords different from those on the archive tape. Specify the update or higher password for the base cluster that the alternate index is being related to. If a physical alternate index is being restored, this parameter must always be specified if the base cluster is password-protected.

Parameters for VSAM AIXS Without DEFER

Specify one or more of the following parameters only if you wish to change a particular value for the cluster from that which is found on the archive medium. If a value is given at both the cluster and component level (for example, ALLOC= and DALLOC=), the value specified at the data or index level will take precedence over the cluster value, just as JSCVSUT does when you issue a DEFINE. Refer to the Hitachi publication Access Method Services Reference for a more complete description of the following parameters.

AIXNAME=

Specify this parameter when you want to restore a specific alternate index from the catalog information archived with the base cluster. The name given on this parameter must be the alternate index name that was present when the base cluster was archived. The DSNNAME parameter must be specified with the base cluster name that contains the definition. The base cluster is NOT restored with this command — only the specific alternate index is. If you are attempting to restore a physical alternate index, it cannot be specified with this parameter (it must be on its own RESTORE command and specified in the DSNNAME parameter). This parameter can be used when you want to restore an alternate index with new attributes. Normally if a base cluster is being restored with its alternate indexes, any parameters defining new attributes are assigned only to the base cluster. The alternate index is restored

with the same attributes it was archived with. If you want to change the alternate index you can specify the following:

```
RESTORE DSN=base.cluster.name,CISZ=4096,  
        NOAIXDEF                      don't define AIX(S)  
RESTORE DSN=base.cluster.name,  
        AIXNAME=aix.name,             alternate index name  
        BUFFER=32000,                 new bufferspace  
        NEW=new.aix.name              new name for AIX
```

AIXDEF

This parameter causes SAMS:Disk to redefine all of the alternate indexes associated with the base cluster when the cluster is restored. This is the default option unless sysparm VSAIXDEF is specified with a value of N.

NOAIXDEF

This parameter causes SAMS:Disk to restore only the base cluster without redefining any of its alternate indexes.

BLDINX

This parameter causes SAMS:Disk to invoke JSCVSUT to rebuild any alternate indexes (BLDINDEX) that SAMS:Disk may define for a base cluster. This is the default option unless sysparm VSBLDINX is specified with a value of N. This parameter is ignored if a physical alternate index is being restored.

NOBLDINX

This parameter causes SAMS:Disk to bypass the call to JSCVSUT to rebuild the alternate index. This parameter is ignored if a physical alternate index is being restored.

Parameters for VSAM Clusters Without DEFER

Specify one or more of the following parameters only if you wish to change a particular value for the cluster from that which is found on the archive medium. If a value is given at both the cluster and component level (for example, ALLOC= and DALLOC=), the value specified at the data or index level will take precedence over the cluster value, just as JSCVSUT does when you issue a DEFINE. Refer to the Hitachi publication Access Method Services Reference for a more complete description of the following parameters.

ALLOC=

This parameter allocates space for the cluster by type, primary amount and secondary amount.

The format is ALLOC=(TYP,PA,SA).

TYP	CYL (cylinders)	TRK (tracks)	REC (records)
PA	Primary allocation amount	Must be from 1 to 9999	
SA	Secondary allocation amount	Must be from 0 to 9999	

FREESPACE=

Specify the percent of space to be left free in control intervals and control areas. Valid values are from 0 through 100, inclusively. The format is

FREESPACE=(CI-percent,CA-percent)

RECSZ=

Specify the average and maximum record size in bytes.

The format is RECSZ=(AVE,MAX).

AVE	Average record size.
MAX	Maximum record size. If left blank, the average record size is assumed to also be the maximum.

BUFFERSPACE=

Specify the amount of buffer space to be provided in bytes. The largest amount is 99999. If a value of 0 is specified, Access Method Services will calculate a new buffer space based on data and index control interval size.

CISZ=

Specify the size of the control interval. The largest CI size is 32768. If this parameter is specified and BUFFERSPACE is not, SAMS:Disk will zero out the buffer space value passed to catalog management so that a new buffer space amount can be calculated based on the new control interval size. If a value of 0 is specified, Access Method Services will calculate a new control interval size.

Caution should be used when restoring an ESDS cluster — changing the CI size on these clusters may destroy internal RBA pointers and cause corruption of the database.

If the cluster was archived in control interval image copy format, changing the control interval size will result in a record length insert error when restoring the cluster. This parameter should not be used in that case.

MASTERPW=

Specify the new master level password to be assigned to the cluster or alternate index.

CONTROLPW=

Specify the new control interval level password to be assigned to the cluster or alternate index.

UPDATEPW=

Specify the new update level password to be assigned to the cluster or alternate index.

READPW=

Specify the new read level password to be assigned to the cluster or alternate index.

ATTEMPTS=

Specify the number of attempts (0 to 7) for entering a password.

OWNER=

Specify the owner ID. It is 1 to 8 alphanumeric or special characters.

MVOLSERS=

Specify a list of from one to five volumes on which the cluster is to be allocated.

FOR=

Specify the retention period of the cluster in days. The valid range is from 0 to 9999.

TO=

Specify the expiration date in Julian format (yyddd).

SHROPTIONS=

Specify how the cluster can be shared among users. The format is SHROPTIONS=(CR,CS).

CR	Cross-region options are 1 through 4
CS	Cross-system options are 3 through 4

KEYS=

Specify the key length and offset in the record for a key-sequenced data set.

The format is KEYS=(LN,OF).

LN	Length of the key
OF	Offset of the key into the record

NOERASE

This parameter indicates the cluster's data component is not to be overwritten with binary zeros when its catalog entry is deleted.

IMBED

This parameter indicates the sequence set of the cluster is to be written with the data component.

NOIMBED

This parameter indicates the sequence set of the cluster is not to be written with the data component.

ORDER

This parameter indicates the volume(s) are to be used in the order listed.

UNORDER

This parameter indicates the volume(s) are to be used in any order.

REPLICATE

This parameter indicates each index record is to be written around a track as many times as it will fit.

NOREPLICATE

This parameter indicates each index record is not to be written around a track as many times as it will fit.

REUSE

This parameter indicates the cluster can be opened again with the high-used RBA set to zero.

NOREUSE

This parameter indicates the cluster cannot be opened again with the high-used RBA set to zero.

SPAN

This parameter indicates the cluster can have records longer than the control interval.

NOSPAN

This parameter indicates the cluster cannot have records longer than the control interval.

SPEED

This parameter indicates the data component is not to be preformatted before records are loaded.

RECOVERY

This parameter indicates the data component is to be preformatted before records are loaded.

UNIQUE

This parameter indicates the cluster's space allocation is to come from the volume's available space, not from a VSAM data space.

SUBALLOC

This parameter indicates the cluster's space allocation is to come from a VSAM data space on the volume.

WRITECHK

This parameter indicates a check is to be issued after every write.

NOWRITECHK

This parameter indicates a check is not to be issued after every write.

Parameters for VSAM Data Components Without DEFER

Specify one or more of the following parameters only if you wish to change a particular value for the cluster from that which is found on the archive medium. If a value is given at both the cluster and component level (for example, **ALLOC=** and **DALLOC=**), the value specified at the data or index level will take precedence over the cluster value, just as JSCVSUT does when you issue a **DEFINE**. Refer to the Hitachi publication *Access Method Services Reference* for a more complete description of the following parameters.

DNAME=

This parameter specifies a new data component name. You may specify either an explicit data set name, or a symbolic name that starts with a "\$". A "\$" instructs SAMS:Disk to use the cluster name as a base name, and to append to it the string

following the “\$”. A second “\$” causes the string “.DATA” to be appended to the cluster name.

For example with a cluster name of VSAM.CLUSTER:

- If DNAME=VSAM.CLSTR.DATA.COMP, the data component name will be VSAM.CLSTR.DATA.COMP.
- If DNAME=\$\$, the data component name will be VSAM.CLUSTER.DATA.
- If DNAME=\$.DATA.COMP, the data component name will be VSAM.CLUSTER.DATA.COMP.
- If DNAME=\$DATA.COMP, the data component name will be VSAM.CLUSTER.DATA.COMP.

DALLOC=

This parameter allocates space for the data component by type, primary amount and secondary amount.

The format is DALLOC=(TYP,PA,SA).

TYP	CYL (cylinders)	TRK (tracks)	REC (records)
PA	Primary allocation amount	Must be from 1 to 9999	
SA	Secondary allocation amount	Must be from 0 to 9999	

Note: SAMS:Disk will specify the values you request, but Hitachi’s catalog management may change it and give absolutely no warning that it has done so. This is usually caused by combinations of attributes that it does not like.

DCISZ=

Specify the size of the control interval in the data component. The largest CI size is 32768. If this parameter is specified and BUFFERSPACE is not, SAMS:Disk will zero out the buffer space value passed to catalog management so that a new buffer space amount can be calculated, based on the new control interval size. If a value of 0 is specified, Access Method Services will calculate a new control interval size.

Caution should be used when restoring an ESDS cluster — changing the CI size on these clusters may destroy internal RBA pointers and cause corruption of the database.

DMASTERPW=

Specify the new master level password of the data component.

DCONTROLPW=

Specify the new control interval level password of the data component.

DUPDATEPW=

Specify the new update level password of the data component.

DREADPW=

Specify the new read level password of the data component.

DATTEMPTS=

Specify the number of attempts (0 to 7) for entering a password for the data component.

DOWNER=

Specify the new owner ID of the data component. It is 1 to 8 alphanumeric or special characters.

DSHROPTIONS=

Specify how the data component can be shared among users.

The format is DSHROPTIONS=(CR,CS).

CR	Cross-region options are 1 through 4
CS	Cross-system options are 3 through 4

DVOLSER=

Specify a list of from one to five volumes on which the data component is to be allocated.

DORDER

This parameter indicates the volume(s) are to be used in the order listed.

DUNORDER

This parameter indicates the volume(s) are to be used in any order.

DWRITECHK

This parameter indicates a check is to be issued after every write.

DNOWRITECHK

This parameter indicates a check is not to be issued after every write.

Parameters for VSAM Index Components Without DEFER

Specify one or more of the following parameters only if you wish to change a particular value for the cluster from that which is found on the archive medium. If a value is given at both the cluster and component level (for example, `ALLOC=` and `DALLOC=`), the value specified at the data or index level will take precedence over the cluster value, just as `JSCVSUT` does when you issue a `DEFINE`. Refer to the Hitachi publication *Access Method Services Reference* for a more complete description of the following parameters.

INAME=

This parameter specifies a new index component name. You may specify either an explicit data set name or a symbolic name that starts with a “\$”. A “\$” instructs `SAMS:Disk` to use the cluster name as a base name and to append to it the string following the “\$”. A second “\$” causes the string “.INDEX” to be appended to the cluster name.

For example with a cluster name of `VSAM.CLUSTER`:

- If `INAME=VSAM.CLSTR.INDEX.COMP`, the index component name will be `VSAM.CLSTR.INDEX.COMP`.
- If `INAME=$$`, the index component name will be `VSAM.CLUSTER.INDEX`.
- If `INAME=$.INDEX.COMP`, the index component name will be `VSAM.CLUSTER.INDEX.COMP`.
- If `INAME=$INDEX.COMP`, the index component name will be `VSAM.CLUSTER.INDEX.COMP`.

IALLOC=

This parameter allocates space for the index component by type, primary amount and secondary amount.

The format is `IALLOC=(TYP,PA,SA)`.

TYP	CYL (cylinders)	TRK (tracks)	REC (records)
PA	Primary allocation amount	Must be from 1 to 9999	
SA	Secondary allocation amount	Must be from 0 to 9999	

Note: `SAMS:Disk` will specify the values you request, but Hitachi’s catalog management may change it and give absolutely no warning that it has done so. This is usually caused by combinations of attributes that it does not like.

ICISZ=

Specify the size of the control interval in the index component. The largest CI size is 32768. If this parameter is specified and BUFFERSPACE is not, SAMS:Disk will zero out the buffer space value passed to catalog management so that a new buffer space amount can be calculated based on the new control interval size. If a value of 0 is specified, Access Method Services will calculate a new control interval size.

IMASTERPW=

Specify the new master level password of the index component.

ICONTROLPW=

Specify the new control interval level password of the index component.

IUPDATEPW=

Specify the new update level password of the index component.

IREADPW=

Specify the new read level password of the index component.

IATTEMPTS=

Specify the number of attempts (0 to 7) for entering a password for the index component.

IOWNER=

Specify the new owner ID of the index component. It is 1 to 8 alphanumeric or special characters.

ISHROPTIONS=

Specify how the index component can be shared among users.

The format is ISHROPTIONS=(CR,CS).

CR	Cross-region options are 1 through 4
CS	Cross-system options are 3 through 4

IVOLSERS=

Specify a list of from one to five volumes on which the index component is to be allocated.

IORDER

This parameter indicates the volume(s) are to be used in the order listed.

IUNORDER

This parameter indicates the volume(s) are to be used in any order.

IWRITECHK

This parameter indicates a check is to be issued after every write.

INOWRITECHK

This parameter indicates a check is not to be issued after every write.

JCL for Processing Queued Requests

To dispose of the deferred restore requests — that is, to actually restore the data sets — the same JCL is used as for immediate restores, with the exception that the module executed is ADSDM279. (Since the input consists of the queued requests, no //SYSIN stream is needed.)

```
//jobname JOB (acct,info),etc.
//DISPOSE EXEC RESTORE,PARM.RESTORE=ADSDM279
```

The completion status for each request processed is maintained within the RESTCMDS subfile, indicating if the request is pending or complete and, if complete, the time of completion.

Note: The MFILES DD statement is not supported in dispose processing. If you have more than one files data set, you must execute DISPOSE separately for each one.

Tape Pull List for Queued Requests

If you use the deferred restore capability, you may also want to generate a list of tapes to be pulled out of the tape library prior to running the job that actually restores the data sets. A utility to create this list is provided.

The following JCL may be used. (It is also supplied as member PULLJCL in the SAMS:Disk installation library.) The only alteration needed is to supply the proper data set names as indicated in the sample JCL.

```
//jobname JOB (acct,info),etc.
//STEP00 EXEC PGM=ADSMI002,PARM=ADSDM603
//STEPLIB DD DISP=SHR,DSN=SAMS.DISK.LOADLIB ADJUST
/* *****
/* * GENERATE TAPE PULL LIST FOR DEFERRED RESTORES *
/* *****
//ABNLDUMP DD DUMMY
//CMDPRINT DD SYSOUT=*
//FILES DD DISP=SHR,DSN=SAMS.DISK.FILES ADJUST
//MSGPRINT DD SYSOUT=*
//PARMLIB DD DISP=SHR,DSN=SAMS.DISK.PARMLIB ADJUST
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//TBLPRINT DD SYSOUT=*
```

Figure 6-2. JCL for Tape Pull List

To generate the list, run this JCL prior to running the dispose job as described above. The only condition this utility does not provide for is the insertion of a DRESTORE request after the running of this job but before the dispose job. In this case, the dispose job may call for a mount of an archive tape that was not in the list.

Condition Codes for Tape Pull List

The following condition codes are set at the completion of the pull list job:

Table 6-2. Tape Pull Condition Codes

Code	Meaning
0	One or more tapes or cartridges are needed
4	No tapes or cartridges are needed

Listing of Deferred Requests

The status of the deferred restore requests may be listed using the following JCL:

```
//jobname JOB (acct,info),etc.
//STEP EXEC LISTREQ
LISTREQ DSNNAME=,RESTORE,ALL
```

DSNAME=

A list of one or more data set names or prefixes for which request entries are to be listed. If this parameter is not specified, the ALL parameter as defined below must be specified. (DSN=/ is equivalent to the ALL parameter.)

RESTORE

The presence of this parameter results in queued DRESTORE request records being selected. It may be abbreviated as RES.

ALL

The presence of this parameter results in all the DRESTORE request records being selected.

Erase a Deferred Request from the Queue

The DERASE command may be used with the following JCL to delete requests from the DRESTORE queue:

```
//jobname JOB (acct,info),etc.  
//STEP EXEC DERASE  
DERASE DSNAME=,RESTORE
```

DSNAME=

A list of one or more data set names or prefixes for which deferred requests are to be deleted. An acceptable abbreviation for this keyword is DSN. A maximum of 20 data set names or prefixes are allowed.

RESTORE

The presence of this parameter indicates that the deferred requests to be deleted are restore transactions. It may be abbreviated as RES.

Implicit Recovery and Volume Defragmentation

An individual data set or cluster is restored via the RESTORE command. A group of data sets, data sets on an entire volume, or a group of volumes, however, are restored via the RECOVER command. Recover processing makes use of all of the applicable RESTORE parameters, plus several additional ones needed to provide greater flexibility unique to the recovery process.

When recovery is being applied to a new volume, or to one or more volumes with sufficient space, the standard benefits provided by individual data set restores are “expanded” to the volume level:

- Multiple data set extents are combined, resulting not only in defragmented data sets, but defragmented volumes as well.
- PDS data sets are compressed.
- VSAM clusters are reorganized.
- Different device types can be used.
- CATALOG entries can be recreated (including VSAM).

The following additional features are unique to RECOVER:

- SIMULATE mode — to see just what data sets will be recovered, based upon the supplied command and parameters.
- VTOC-based recovery — limiting data sets to those listed in the most recent backup copy of the volume’s VTOC.
- Date and Time recovery - telling recovery to use only those backups that were taken within a specified time period.
- Backups-only selection — skipping the restores for those data sets that were scratched when placed in the SAMS:Disk archives.
- Non-VSAM and VSAM-only separation — the ability to process these two distinctly different types separately if desired. Default processing will recover both types.

Note: If an OS CVOL or a VSAM DMF/EF Type-2 catalog resides on a volume being recovered and catalogs are selected for processing, they will be restored first. This permits any catalog updates done for the remaining recovered data sets to be properly entered. See below for detailed information regarding the recovery of DMF/EF Type-2 catalogs.

- Restart capability

- Implicit selection based upon fields unique to the DSNINDEX record.

The following notes pertain to VSAM cluster recovery only:

- Clusters may be moved freely between DMF/EF Type-2 and non-DMF/EF Type-2 catalogs
- VSAM KSDS cluster names do not appear in the VTOC—their associated data and index component names appear instead. In addition, suballocated (non-unique) VSAM clusters do not appear in the VTOC. Since SAMS:Disk processes based on the cluster name, VTOC-based recovery will not work for them. For volumes with VSAM clusters, use the DATE=, TIME=, EDATE= and ETIME= parameters instead of VTOC, or use two RECOVER commands, one with VTOC and the other with VSAMONLY, or do a RECOVER VTOC and then explicitly restore the needed clusters.
- When recovering suballocated (non-unique) VSAM clusters to a new volume, the user must predefine the VSAM space as desired. Once this space is created, SAMS:Disk will dynamically define (redefine) the VSAM clusters into it. Unique clusters will have their self-containing spaces defined automatically by SAMS:Disk.
- Predefined (preallocated) VSAM clusters are permitted only under certain conditions. See the discussion above for detailed information.
- TOCAT= specification — used to specify the name of a VSAM catalog in which the restored clusters are to be defined. For VSAM catalogs only, the named catalog must own the volume to which the clusters are being restored.

Note: This parameter is normally not needed, assuming that the master catalog contains the correct alias entries to cause it to select the proper catalog. If it does not, a STEPCAT should be provided, in which case this parameter still isn't needed.

- CATALOGS=, DEFALIAS= and EXPORTF= parameters on the SELECT command pertain solely to the recovery of DMF/EF Type-2 catalogs.

RECOVER Command

The SAMS:Disk recover function (executed under the FMS PROC discussed on page 77), allows you to restore data sets and OS catalogs without the need for pre-allocation. You may also restore VSAM clusters and DMF/EF Type-2 catalogs. In most cases, you can even restore to a device type other than that from which the data set was backed up or archived. The following is a list of the data sets supported:

- Physical Sequential (PS)
- Direct Access (DA)
- Indexed Sequential (IS)
- Partitioned Organization (PO and POE)
- Access Method—VSAM (AM)

Commands	JCL PROC
Available:	Executed:
RECOVER	FMS

General Information

The recover function begins by filtering all of the candidate data sets and sequencing them by file number within each archive tape (or archive data set on disk), the optimum performance arrangement. All data sets being restored from the archives on disk are done before those archived to tape.

Selection Criteria

All applicable Selection Criteria are documented under the topic "*Criteria Statement Operands*" beginning on page 87, and can be used to restrict Recover to any combination of selection criteria. Please review that section before attempting a recover.

Multiple Files Data Set for Recover

Multiple files data sets are supported for Recover processing. If your installation maintains multiple files data sets and you want to make them all available to the Recover function, you should use the MFILES DD statement instead of the FILES DD statement. Add each files data set name to the MFILES concatenation, up to 256 files data sets are supported. The order of the concatenation determines the order of search for the data set. The first files data set that contains an index record for the archived data set will be used, even if another files data set contains a more recent version of the data set. Therefore, it is very important that files data sets be specified in the proper order.

Note: The FILES DD statement will be ignored if there is also an MFILES DD statement in the procedure. Also note that concatenated files data sets are not supported with the FILES DD statement.

RECOVER Condition Codes

Condition codes received from the FMS PROC are described in the Overview section. Please turn to page 78 for details.

SYNTAX

The syntax of the RECOVER command is as follows:

```
RECOVER TOVOLUME=, POOL=, DISP=, NEWHLQ=, SCRATCH,
        CREATE, ERASE, NOLOAD, AIX=, TOCAT=,
        EXPORTF=, DEFALIAS=
```

TOVOLUME=

By default, the recovery of data set(s) is to the volume(s) from which the data set(s) were backed up or archived. This parameter allows for an alternate target volume to be used. A maximum of one explicit entry is supported.

POOL=

This parameter may be used to specify the name of a DASD Pool (a list of target volumes) which RECOVER can select from. This parameter is limited to only non-VSAM data sets, and is mutually exclusive with the TOVOLUME= parameter. For information on how to setup DASD Pool support, see the sysparm description of DFLTPOOL on page 131, and the PARMLIB description of DASDPOOL on page 444, both in the *Systems Guide*.

DISP=

This parameter indicates to SAMS:Disk what the final catalog disposition of the restored non-VSAM data set(s) should be. VSAM data sets will always be cataloged after restore, regardless of the value specified for this parameter. Three values are supported:

1. **CATALOG** — This is the default value, and will cause all recovered data sets to be cataloged. If a recovered data set is already cataloged, the existing catalog entry is left unaltered.
2. **NOCATALOG** — This non-VSAM only value prevents RECOVER from cataloging any data sets to the target volume.
3. **RECATALOG** — This non-VSAM only value forces RECOVER to catalog the data sets to the target volume. This value will force the uncatalog of existing data sets by the same name. For related information, see the sysparm description of RESRCAT on page 159 of the *Systems Guide*.

NEWHLQ=

This parameter forces a rename of all recovered data sets. Each data set recovered will have its High Level Qualifier changed to the value of this parameter.

SCRATCH

Indicates that if the current data set being processed already exists on the target volume, scratch it prior to recovery. For related information, see the sysparm description of RESRCAT on page 159 of the *Systems Guide*.

CREATE

This parameter causes SAMS:Disk to set the creation date in the Format-1 DSCB for each recovered data set to the current date. Default processing leaves the original creation date unaltered.

ERASE

When recovering a PO data set to a preallocated PO, the presence of this parameter erases the contents of the target data set with the restored version. If ERASE is not specified when recovering to an existing PO, sufficient space must be available for RECOVER to "merge" the restored members into the data set.

For related information, please turn to the sysparm description for *PREALLOC* on page 156 in the *Systems Guide*.

NOLOAD

For non-VSAM data sets, this parameter causes SAMS:Disk to allocate the data sets being recovered onto the target volume, but not load any of their records. For VSAM data sets, this parameter causes SAMS:Disk to define the clusters being recovered onto the target volume, but not load any of their records.

AIX=

This parameter indicates how alternate indexes are to be processed for the clusters being recovered. Three values are supported:

1. **YES** — This is the default value, and will cause all clusters being recovered to have their associated alternate indexes defined and rebuilt.
2. **DEFINEONLY** — Causes all clusters to have any associated alternate indexes defined, but not rebuilt. This parameter may be abbreviated DEF.
3. **NODEFINE** — Causes all clusters that have associated alternate indexes to not be defined or rebuilt. This parameter may be abbreviated NO.

TOCAT=

Specifies the catalog where the recovered cluster should be defined. It applies to VSAM only, and overrides the catalog selected by normal catalog management facilities.

EXPORTF=

Indicates whether or not SAMS:Disk should recover DMF/EF Type-2 catalogs. Two values are supported:

1. **NO** — This is the default value, and will recover (IMPORT) DMF/EF Type-2 catalogs directly (i.e., recreates the catalog)
2. **YES** — This option causes SAMS:Disk to create a sequential file in “Export-Temp” format, allowing you to execute the IMPORT function as a separate job. For related information, please turn to the sysparm descriptions for *CATBKNAM*, *CATBKSPC*, and *CATBKUNT*, beginning on page 125 of the *Systems Guide*.

DEFALIAS=

Indicates whether or not SAMS:Disk should redefine alias entries into the master catalog. Two values are supported:

1. **YES** — This is the default value, and causes alias entries for the catalog being recovered to be redefined into the master catalog.
2. **NO** — Prevents the alias entries from being redefined.

Note: This parameter is ignored when EXPORTF=YES is specified.

Examples

Various SAMS:Disk executions have been developed and grouped together in a separate chapter for easy reference. Several recovery scenarios executing the FMS PROC have been added for this release, beginning on page 436.

Volume Defragmentation

A common, though infrequent, task at many locations is to “defrag” a volume, which at minimum usually means to consolidate the used tracks on the pack, such that free space is in just one or two extents. To SAMS:Disk, it means the following:

1. Consolidate used tracks to obtain large contiguous free space extents.
2. Combine multiple extent data sets into a single extent.
3. Compress PDS data sets.
4. Reorganize VSAM clusters.
5. Compress the VTOC.

A recommended approach to doing this with SAMS:Disk is as follows:

1. Allocate and format a new “mini” files data set.
2. Establish a JCL procedure that formats this files data set and then uses it in a DSCL run that archives and scratches all data sets on the pack. Follow this with a RECOVER execution to restore all of the data sets on the volume.

The advantage of using a separate files data set for this special function is that it keeps these reorganization index entries separate from the archive/backup entries, and speeds the time needed to process the files data set, since it contains fewer entries.

When completed, ARCHVOLS used in the temporary files data set can be deleted from your tape management system by using the following IXMAINT command:

```
PURGE DSN=/ ,VOL=/
```

Sample Reports

RESTORE

1987.337 DEC 03, 1987	R E S T O R E D D A T A S E T S												PAGE 1
THURSDAY 4.32 PM													SAMS:Disk 8.2
DATA SET NAME / NEWNAME	SOURCE VOLUME	DEVICE TYPE	ARCHIVED TIME DATE	TARGET VOLUME	DEVICE TYPE	DS CAT	REC ORG FM	BLKSZ	LRECL	TYPE	ALLO ALLOC	---TRACKS--- IDLE	CREDAT
LABS.DMSDLG.LOADLIB	LABS83	3380	1631 1987.337	LABS83	3380	Y PS	FB	4080	80	TRK	2 1	87.334	
LABS.DMSDLG.FILES	LABS83	3380	1631 1987.337	LABS83	3380	Y PS	FB	4080	80	TRK	2 1	87.334	
LABS.DMSDLG.CNTL	LABS83	3380	1631 1987.337	LABS83	3380	Y PS	FB	4080	80	TRK	2 1	87.334	
LABS.DMSDLG.SRC	LABS83	3380	1631 1987.337	LABS83	3380	Y PS	FB	4080	80	TRK	2 1	87.334	
	TOTAL DATASETS		TOTAL TRACKS										
	-----		-----										
RESTORED...	4		8										

Figure 6-3. Sample Restore Report

RECOVER

1996.191 JUL 09, 1996	R E S T O R E D D A T A S E T S												PAGE 2
TUESDAY 9.30 AM													SAMS:DISK 9.0.0
DATA SET NAME / NEWNAME	SOURCE VOLUME	DEVICE TYPE	ARCHIVED TIME DATE	TARGET VOLUME	DEVICE TYPE	DS CAT	REC ORG FM	BLKSZ	LRECL	TYPE	ALLO ALLOC	---TRACKS--- IDLE	MPFILES CONCAT
QDMSTEST.DSCR001.R820.PROCLIB	FRGMA	3380	1121 1996.069	FRGMA	3380	Y PO	FB	27920	80	BLK	110 100	1	
QDMSTEST.DSCR001.R820.SPFTST2	FRGMA	3380	1121 1996.069	FRGMA	3380	Y PO	FB	4080	80	BLK	3 2	1	
QDMSTEST.DSCR001.R820.TERM	FRGMA	3380	1121 1996.069	FRGMA	3380	Y PO	FBA	6050	121	BLK	2 1	1	
QDMSTEST.DSCR001.SSM.V211.LOAD	FRGMA	3380	1121 1996.069	FRGMA	3380	Y PO	U	6144	0	BLK	56 4	1	
QDMSTEST.DSCR001.USE.PDSTEST	FRGMA	3380	1121 1996.069	FRGMA	3380	Y PO	VBA	6144	125	CYL	3090 1353	1	
QDMSTEST.DSCR001.V100BASE.BTLIST	FRGMA	3380	1121 1996.069	FRGMA	3380	Y PO	FEM	3630	121	TRK	217 44	1	
QDMSTEST.DSCR001.V100BASE.BTLOADVS	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	FB	3200	80	TRK	20 0	1	
QDMSTEST.DSCR001.V100BASE.BTOBJVSE	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	FB	3200	80	TRK	13 2	1	
QDMSTEST.DSCR001.V100BJD.BTLOADVS	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	FB	3200	80	TRK	24 5	1	
QDMSTEST.DSCR001.V100BJD.BTOBJSPE	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	FB	3200	80	TRK	37 11	1	
QDMSTEST.DSCR001.V100BJD.BTOBJVSE	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	FB	3200	80	TRK	4 1	1	
QDMSTEST.DSCR001.V100REL.BTASM	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	FB	3120	80	TRK	3 1	1	
QDMSTEST.DSCR001.V100REL.BTLIST	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	FEM	3630	121	TRK	4 1	1	
QDMSTEST.DSCR001.V100REL.BTMAC	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	FB	3120	80	TRK	3 1	1	
QDMSTEST.DSCR001.V100REL.BTOBJSPE	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	FB	3200	80	TRK	8 2	1	
QDMSTEST.DSCR001.V100REL.BTOBJVSE	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	FB	3200	80	TRK	3 1	1	
QDMSTEST.DSCR001.V100TEST.BTASM	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	FB	3120	80	TRK	2 0	1	
QDMSTEST.DSCR001.V100TEST.BTH	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	FB	3120	80	TRK	3 1	1	
QDMSTEST.DSCR001.V100TEST.BTLNK	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	FB	3120	80	TRK	3 1	1	
QDMSTEST.DSCR001.V100TEST.BTOBJSPE	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	FB	3200	80	TRK	10 2	1	
QDMSTEST.DSCR001.V100TEST.BTOBJVSE	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	FB	3200	80	TRK	10 2	1	
QDMSTEST.DSCR001.V211.LOAD	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	U	6144	0	BLK	16 14	1	
QDMSTEST.DSCR001.V500BJD.STDEBUG	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	U	4080	0	TRK	27 13	1	
QDMSTEST.DSCR001.V500GRL.STMAC	FRGMA	3380	1122 1996.069	FRGMA	3380	Y PO	FB	3120	80	TRK	2 1	1	
TOTAL DATA SETS	74		TOTAL ALLOCATED TRACKS		8449								
			TOTAL IDLE TRACKS				3085						

Figure 6-4. Sample Recover Report

Chapter 7. VBACKUP/VRECOVER

Typical SAMS:Disk processing operates at the data set level; that is, each data set is processed as an individual logical unit, with the data being read and written in logical order. However, in order to provide the capability to perform Stand—Alone Restore, SAMS:Disk has developed the ability to process at the volume level for backup and recover processing; that is, a DASD volume is read and written starting at cylinder 0 track 0, and continuing through the end of the used cylinders.

These functions provide additional versatility to SAMS:Disk in an online environment, and are documented in this section. For further information on Stand—Alone Restore, see the *Stand—Alone Restore Guide*.

COMMAND SUMMARY:

VBACKUP
VRECOVER

PROGRAM EXECUTED:

ADSST001
ADSDM330

SPECIAL PARMLIB MEMBERS:

none

SPECIAL FILES:

none

PROCEDURE(S) AVAILABLE:

- RECOVER
- DMS

General Information

Overview of Volume-Level Processing

Typical SAMS:Disk processing operates at the data set level; that is, each data set is processed as an individual logical unit, with the data being read and written in logical order. However, in order to provide the capability to perform Stand—Alone Restore, SAMS:Disk has developed the ability to process at the volume level for backup and recover processing; that is, a DASD volume is read and written starting at cylinder 0 track 0, and continuing through the end of the used cylinders.

The volume-level backup function (VBACKUP) is invoked through DSCL. It will produce a backup tape or tapes containing an entire DASD volume, starting at cylinder 0, track 0 and continuing through all used cylinders.

The volume-level recovery function is invoked under the RECOVER procedure but through the VRECOVER command. When specified at recover time, this function uses the volume-level backup tape or tapes to overlay the entire DASD volume in extent track order, from cylinder 0 to the end of the used cylinders.

Stand—Alone Restore is a program designed to restore a DASD volume without the presence of the VOS3 operating system. It runs under the Hitachi/M-series and Hitachi/ES instruction sets and runs under VMS as a guest operating system. This function is available as a separate product, and is documented in the *Stand—Alone Restore Guide*.

When the Stand—Alone Restore function is performed, SAMS:Disk uses the volume-level backup tape or tapes in basically the same fashion as they are used for an online volume-level recovery. That is, they are used to overlay either all the tracks on the pack or a subset of the tracks as specified by absolute track addresses. For further information, see the *Stand—Alone Restore Guide*.

Although volume-level processing was developed to provide backup tapes that can be used for Stand—Alone Restore, these backups also provide additional versatility within normal SAMS:Disk storage management tasks.

VBACKUP Function

The SAMS:Disk volume-level backup function is invoked through DSCL. The volume-level backup function will create one backup tape (or a set of backup tapes) for each volume. Each tape or set of tapes will contain an entire DASD volume, starting at cylinder 0, track 0, and continuing through all used cylinders.

Because this is a full-volume backup function, change-bit actions are not observed, and DSNINDEX records are not written for the data sets on the volume.

SAMS:Disk will request a new backup tape automatically if the backup processing switches to a new DASD volume. That is, if the volume-level backup includes two or more DASD volumes, the contents of the volumes are not mixed on the backup tapes.

The volume-level backup will look like one large data set to SAMS:Disk. It will have one DSNINDEX record with an internally generated data set name of

DMSOS.Vxxxxxxx

where “xxxxxx” is the volume serial. The DSNINDEX record points to the ARCHVOLS records which identify the tape volumes used for the backup.

SAMS:Disk generates data set names for the volume-level backup tapes in the same manner as it generates names for the data set-level backup tapes created by ARCHIVE/BACKUP; that is, sysparms ARC0NAME and ARCCNAME provide the base-level name.

The DSNINDEX records for volume-level backups are flagged to indicate an extent track image backup with an x’01’ in the second LISTD flag. The ARCHVOLS records for volume-level backups are flagged with an x’10’ in the second LISTV flag.

If you have instructed SAMS:Disk to use data compression during backup processing (sysparm DCDATACP specified with a value of Y), volume-level backup processing will either use the SAMS:Disk-supplied techniques 0 or 2. That is, to ensure data accessibility at Stand—Alone Restore time, any other compression technique for normal SAMS:Disk processing will be ignored.

Track Overflow Data Sets

Because VBACKUP uses the EXCP access method, it is not able to back up track overflow data sets. If SAMS:Disk encounters a track overflow data set on the volume you are backing up during VBACKUP processing, it will bypass the data set, issue a message and continue processing.

If the track overflow data set is updated following the most recent VBACKUP run, it can be restored during VRECOVER if the UPDATE parameter is specified. However, if the data set has not been updated since the last VBACKUP run, it will

not be recovered during VRECOVER processing and must be restored explicitly from a previous data set-level backup copy.

VBACKUP Security Considerations

SAMS:Disk performs all volume-level security checking prior to backing up each DASD volume. No data set-level security checking is performed. If there is not sufficient authority to backup a volume, the VBACKUP job will end and the volume will not be backed up. SAMS:Disk will only check for volume-level authority to the volume if your security interface is activated and sysparm SECURVOL defaults to a value of Y. Use of the VBACKUP command itself can be restricted by SAMS:Disk.

Data Integrity Considerations

If SAMS:Disk encounters any physical I/O errors during the course of the volume-level backup, the CCHH for the track is saved. At the end of the backup processing, SAMS:Disk writes a record to the archive tape that contains the CCHH for the tracks in error. The number of I/O errors allowed before the volume is bypassed is determined by the sysparm IOERRLIM. For detail information, please turn to page 146 in the *Systems Guide*.

This record of I/O errors is prefixed with a record type of “I”. The DSNINDEX record Flag2 field is set to x’08’, indicating a data set has been backed up with I/O errors.

ARCHIVE Management Considerations

The tapes SAMS:Disk creates for volume-level backup are exempt from MERGE and EXTRACT processing. They are handled in the same manner as normal SAMS:Disk data set-level backup tapes for the IXMAINT, REBUILD, TAPE-COPY, and REPARCH functions.

VBACKUP Command and Parameters

The control statements for a volume-level backup are:

```
VBACKUP DSNENQ= , RESERVE= , RETPD= , EXPDT= , RPT= , LSTAR
```

DSNENQ=

This parameter defaults to SHR, which will issue an enqueue for every data set on the volume. If SAMS:Disk cannot obtain an enqueue, it will continue processing, but will issue a message to identify the data set, and set the Flag1 field in the DSNINDEX record X'01' to indicate that the volume was backed up without complete enqueue integrity.

If this parameter is set to NONE, SAMS:Disk will not issue enqueues for the data sets, but will set the Flag1 field in the DSNINDEX record to X'01' to indicate the lack of integrity.

RESERVE=

This parameter defaults to YES, which causes SAMS:Disk to issue a hardware reserve and VTOC enqueue. If this parameter is set to NO, SAMS:Disk will not issue the hardware reserve and the VTOC enqueue, and the DSNINDEX record Flag1 field is flagged as X'01' to indicate the lack of integrity.

RETPD=

This parameter and the EXPDT parameter are mutually exclusive and have the same meaning as the identical parameters used for data set-level processing.

Specify a one- to five-digit number to indicate a retention period to be assigned to the volume-level archive data sets. If this parameter is omitted, a default retention period value is obtained from sysparm RETRETPD.

Note: The expiration date for the ARCHVOL is determined by the sysparm *DYNEXPDT*, described on page139d of the *Systems Guide*.

EXPDT=

This parameter and the RETPD parameter are mutually exclusive and have the same meaning as the identical parameters used for data set-level processing.

Specify an expiration date in an accepted SAMS:Disk format to designate the period of time the data set will remain archived. If this parameter is omitted, a default retention period value is obtained from sysparm RETRETPD.

Note: The expiration date for the ARCHVOL is determined by the sysparm *DYNEXPDT*, described on page139d of the *Systems Guide*.

RPT=

Specify this parameter with a value of MAP to produce a report for each DASD volume processed. This report will include information such as data set names and extents. If you specify a value of NONE, which is the default value, no report will be produced.

LSTAR

If this parameter is supplied, the volume-level backup processing will include only the used tracks in physical sequential and partitioned data sets, as defined by the LSTAR value in each data set's format-1 DSCB.

If this parameter is not supplied, the volume-level backup processing will include all allocated tracks on the DASD volume.

VRECOVER Function

The SAMS:Disk volume-level recovery function will overlay the entire volume in extent track order, or a specified range of absolute tracks on the volume. It is invoked from RECOVER processing through the VRECOVER command.

VRECOVER uses the volume-level backup data set identified in the DSNINDEX as DMSOS.Vvolser. This name is generated automatically by SAMS:Disk.

A volume-level recovery must use a target volume that is the same device type as that from which the volume-level backup was taken. However, a single-density 8598 can be recovered to a double- or triple-density 8598, and a double-density 8598 can be recovered to a triple-density 8598. In such cases, the VTOC will be updated to invoke system routines to recalculate the free space.

If the data on the volume-level backup is compressed, it will be decompressed during VRECOVER processing using the SAMS:Disk-supplied techniques.

Note: See “*Track Overflow Data Sets*” on page 253 for a description of alternate processing required to restore these data sets.

If you use the UPDATE parameter on the VRECOVER command and there is a catalog on the volume being recovered, specify the sysparm CATBKDEL with a value of Y. This allows the most current version of the catalog to be restored.

Security Considerations

SAMS:Disk performs all volume-level security checking prior to recovering each DASD volume. No data set-level security checking is performed. If there is not sufficient authority to a volume, the VRECOVER job will end and the volume will not be recovered. SAMS:Disk will only check for volume-level authority to the volume if your security interface is activated and sysparm SECURVOL defaults to a value of Y.

If SECURVOL is specified with a value of N, even though the volume may be protected by your security system, SAMS:Disk will still recover to that volume. This applies to both the target volume and the volume name specified for the IDVOL= parameter of the VRECOVER command. Use of the VBACKUP and VRECOVER commands themselves can be restricted by SAMS:Disk.

VRECOVER Command and Parameters

The command and parameters for a volume-level recovery are:

```
VRECOVER SIM,VOLUMES=,TOVOLUME=,VERSION=,TIMEDATE=,UPDATE,
        DSNENQ=,RESERVE=,GOAHEAD=,IDVOL=,CYL0TRK0=,
        SECERASE,ABSTRKS=,RPT=
```

SIM

This parameter causes processing to appear to take place; however, no data sets will be altered. Messages will be issued as if normal volume-level recover processing were taking place.

VOLUMES=

Specify for this required parameter a list of one or more volumes to be recovered. (A maximum of 50 entries is supported). If you specify more than one volume for this parameter, you may not specify the TOVOLUME=, IDVOL= or ABSTRKS= parameters on the same VRECOVER command.

TOVOLUME=

The volume to which the volume-level recover is directed. If this parameter is not specified, SAMS:Disk will use the volume(s) specified in the VOLUMES= parameter. If the VOLUMES= parameter specifies more than one volume and this parameter is also specified, the command will be rejected.

VERSION=

The VERSION= parameter is mutually exclusive with the TIMEDATE= parameter, and has the same meaning as defined for the RESTORE command.

TIMEDATE=

The TIMEDATE= parameter is mutually exclusive with the VERSION= parameter, and has the same meaning as defined for the RESTORE command.

UPDATE

This optional parameter indicates that any data set changed after the volume-level backup being used for this VRECOVER was taken will subsequently be updated (recovered) from its data set-level backup; that is, recover the data set from the incremental backup copy after the volume-level recovery is complete. This is accomplished by SAMS:Disk internally generating and executing the data set-level RECOVER command. The parameters that are generated for the RECOVER command are:

```
RECOVER VOL=,TOV=,DATE=,TIME=,SCRATCH,NOC,(SIM)
```

The parameter values generated for VOL= and TOV= are the same as those specified on the VRECOVER command. SAMS:Disk obtains the DATE= and TIME= values from the DSNINDEX record of the volume-level backup being used. If the

VRECOVER command is submitted with the SIMULATE parameter specified, the RECOVER command will be generated in SIMULATE mode as well.

By generating this RECOVER command, SAMS:Disk allows you to first recover the entire volume from a volume-level backup, then follow immediately with updates from incremental backups. SAMS:Disk will select the data sets needing to be restored based on the DATE and TIME values generated on the RECOVER command. To be effective, your installation should schedule volume-level backup runs to follow incremental backups. This is because the incremental backup turns off the change bit for updated data sets, thus reducing the number of data sets needing to be restored after the volume-level recover is complete.

DSNENQ=

The DSNENQ= parameter has a default value of EXC, which causes SAMS:Disk to issue an exclusive enqueue for every data set on the volume. If you specify a value of NONE, no enqueues are issued for the data sets. A value of SHR causes shared enqueues to be issued.

RESERVE=

The RESERVE= parameter defaults to YES, and issues a hardware reserve and VTOC enqueue. If you specify a value of NO, no hardware reserve or VTOC enqueue are issued.

Note: Using the defaults for the DSNENQ= and RESERVE= parameters will prevent any updates to the volume during VRECOVER processing.

GOAHEAD=

This parameter controls processing only when an enqueue cannot be obtained or if the hardware reserve fails. The default value is OPERATOR, which causes a message to be issued on the console notifying the operator of the enqueue or reserve that failed, and giving the operator the options to LIST, CONTINUE or CANCEL. If the operator responds "LIST," SAMS:Disk will display at the operator's console a list of the data sets for which the enqueue or reserve failed.

If you specify a value of CONTINUE on this parameter (or if the operator replies "CONTINUE"), SAMS:Disk will continue processing. If you specify a value of CANCEL (or if the operator replies "CANCEL"), SAMS:Disk will issue a message and terminate VRECOVER processing.

IDVOL=

The IDVOL= parameter controls the volume name SAMS:Disk assigns to the target volume. The default value (not specifying this parameter) causes SAMS:Disk to retain the original name of the volume. If you specify a different volume name for this parameter, SAMS:Disk will rename the target volume. (This is sometimes referred to as "clipping" the volume.)

This parameter applies only to a full volume-level recovery; that is, it is mutually exclusive with the ABSTRKS= parameter described below.

CYL0TRK0=

The CYL0TRK0= parameter controls the recovery of the data records, if present, on cylinder 0, track 0. This track is a reserved track on each volume containing a pointer to the VTOC, the volume serial, and other optional data. It often includes IPL text.

When a volume-level recovery is performed, the pointer to the VTOC is always updated to reflect its location on the recovered volume. The default value of YES for this parameter causes all data records on this track to be recovered from the backups. If you specify a value of NO, only the VTOC pointer is updated (and the volume serial, if it is being renamed).

This parameter applies only to a full volume-level recovery; that is, it is mutually exclusive with the ABSTRKS= parameter described below.

SECERASE

If you specify this simple parameter, SAMS:Disk will zero all tracks on the target volume not recovered.

This parameter applies only to a full volume-level recovery; that is, it is mutually exclusive with the ABSTRKS= parameter described below.

ABSTRKS=

This parameter may be specified to overlay only a range of absolute tracks rather than the full volume. ABSTRKS= is mutually exclusive with the UPDATE, ID-VOL=, CYL0TRK0= and SECERASE parameters. If the VOLUMES= parameter specifies more than one volume and this parameter is also specified, the command will be rejected.

When this parameter is specified, the DSNENQ processing is also bypassed.

Specify this parameter in the following format:

```
ABS=( cccchhhh-ccccchhhh,ccccchhhh-ccccchhhh, . . . )
```

Each pair indicates a beginning and ending range of tracks in absolute address format. The range of tracks will be recovered without regard to any data set checking.

If cylinder 0 track 0 is one of the absolute tracks to be recovered, the TO-VOLUME= volume serial is placed in the volume serial field before the remainder of the track is written.

RPT=

This parameter may be specified with a value of MAP to cause SAMS:Disk to produce a VTOC MAP report at the end of the volume-level recovery. If the default value NONE is specified or the Simulate parameter is present, no report is produced.

SAMPLE REPORT

1988.286	OCT 12, 1988	D A T A S E T	E X T E N T	R E P O R T	F O R	V O L U M E	MVS001	PAGE	1
WEDNESDAY	5.19 PM							SAMS:Disk	8.2
DATA SET NAME		DSO	BEG CCHH	END CCHH					
GIM.SGIMLMD0		PO	019E0000	019F001D					
			01F40000	01F4001D					
			01F50000	01F5001D					
			01FB0000	01FB001D					
GIM.SGIMMSG0		PO	01A00000	01A00012					
GIM.SGIMPNL0		PO	01A10000	01A3001D					
			018A0000	018A001D					
			01FA0000	01FA001D					
			02290000	0229001D					
			022A0000	022A001D					
GIM.SGIMPRC0		PO	01A40000	01A40018					
GIM.SGIMTBL0		PO	01A50000	01A50001					
IPO1.CMDPROC		PO	015C0000	015C000B					
IPO1.GENLIB		PO	015F0000	0164001D					
IPO1.HELP		PO	015E0000	015E0003					
IPO1.INSTLIB		PO	01670000	016F001D					
			01FC0000	01FC001D					
			01FD0000	01FD001D					
			01FE0000	01FE001D					
			01FF0000	01FF001D					
			02000000	0200001D					
			02040000	0204001D					

Figure 7-1. VBACKUP/VRECOVER Report

1.

Chapter 8. XCOPY

The SAMS:Disk Disaster Recovery Extract Utility (XCOPY) creates ARCHVOLS from SAMS:Disk archive and backup tapes. With its unique design, XCOPY creates ARCHVOLS in a way that will not impact your normal production backup window. These newly created ARCHVOLS are identical in file format to their archive and backup counterparts, they just reside in a separate files data set.

XCOPY provides its own set of flexible commands and parameters that allow Storage Administrators to copy only a subset of archive and backup data onto the new ARCHVOLS. Data sets can be selected by fully qualifying their data set names, or by selecting a special character and specifying a pattern of a data set name.

SAMS:Disk data compression, activated via the DCDATACP sysparm, has no effect on XCOPY. Data is processed identically as the input volume, whether it was compressed or not.

Finally, XCOPY has an internal mechanism that gives the Storage Administrator the ability to maintain different DSNINDEX and ARCHVOL expiration dates from those on their primary archive and backup copies.

Overview

XCOPY was designed to be an ideal complement to a Data Center's disaster recovery procedure. While running XCOPY, you will benefit in lower CPU usage compared to that of your normal archive and backup production jobs. XCOPY is designed to copy data from an archive and backup ARCHVOL onto a Disaster Recovery ARCHVOL, eliminating all data set and volume selection overhead.

Limitations

This feature does not support archive and backup ARCHVOLS created prior to or equal to SAMS:Disk release 7.8.8.

The Merge flag is set for all XCOPY-created ARCHVOLS. Merge can exclude these XCOPY-created tapes by using the NOMERGEIN parameter.

Basic Operation

XCOPY uses archive and backup ARCHVOLS previously created by SAMS:Disk and the files data set they reside in as input to create subsets of the data. It was developed for Data Centers that were required to create subsets of their backup data to take offsite for disaster recovery. By creating a disaster recovery ARCHVOL from a primary archive and backup ARCHVOL, copying the data can be run outside the backup time window without having to quiesce systems relying on the data.

Automatic Selection

XCOPY is an intelligent process that can automatically select **only** those primary archive and backup ARCHVOLS that it has not yet previously processed. This feature is controlled by the FLAG and CHKFLAG parameters. After an archive and backup ARCHVOL has been processed, its marked as “XCOPY processed” and may be automatically skipped during subsequent XCOPY runs.

This feature allows you to set up XCOPY to run regularly and automatically generate a list of input ARCHVOLS of disaster recovery data. It will find new backup data taken during normal backup processing, select data sets to be copied for backup data sets, and copy the data.

Secondary Files Data Set

XCOPY can be optionally set up to record all backup information into a separate files data set. Input index records are selected and located through the FILES DD statement, while output index records are recorded in the files data set through the OFILES DD statement. The data set allocated to the OFILES DD statement has the same format as the data set allocated to the FILES DD statement, and may be used by any other SAMS:Disk function that normally uses the files data set.

The advantage of having a secondary files data set for disaster recovery is that it simplifies the recovery process. This files data set and all corresponding ARCHVOLS may be taken to a new location and used for individual data set restoration, or for full system recovery. Finally, by having a files data set that only contains information about offsite data, SAMS:Disk will never attempt to restore data from an ARCHVOL outside the set of offsite ARCHVOLS.

Expiration Date Support

You can optionally assign special expiration dates to the disaster recovery ARCHVOLS by supplying either the RETPD= or EXPDT= parameter on the XCOPY command. When not specified, the expiration dates of the output ARCHVOL will match that of the input ARCHVOL.

Condition Codes

The following condition codes are returned from the stepname XCOPY during the XCOPY process:

Table 8-1. Condition Codes and Descriptions

Code	Description
0	Successful execution
1	Successful execution (extra copy step needed)
4	Warning messages issued
5	Warning messages issued (extra copy step needed)
8	Error terminated processing
9	Error terminated processing (extra copy step needed)
16	Critical error caused XCOPY to terminate

Implementation

Create a Secondary Files Data Set

Using normal SAMS:Disk procedures, create a secondary files data set for recording disaster recovery backup information. Detailed instructions to accomplish this task can be found under the topic *"Formatting the Files Data Set"* beginning on page 7 of the *Systems Guide*.

Identify Data Sets for Disaster Recovery

Prepare a list of data set names and patterns to specifically identify those data sets to be included in the disaster recovery set of data. This list of data sets will be used as selection criteria for selecting data sets from the input files data set.

Determine Expiration Criteria

Using normal SAMS:Disk index maintenance planning, determine when SAMS:Disk should expire disaster recovery data.

Running XCOPY

XCOPY uses the XCOPY PROC which contains the basic JCL needed to run the utility. In addition to executing the PROC, you will need to supply the following JCL items:

1. Input files data set
2. Secondary files data set
3. System Parameter Specifications
4. SET MODE=SIMULATE (optional)
5. SELECT= command to select the archive and backup data sets to be copied
6. XCOPY command

System Parameter Considerations

The following system parameters are used to define output device types and ARCHVOL naming conventions. Detailed descriptions of these system parameters can be found beginning on page 149 in the *Systems Guide*.

- MERP1NAM
- MERP1TYP
- MERC1NAM
- MERC1TYP

Simulate Mode Considerations

By using the SET MODE=SIMULATE command in an XCOPY job, SAMS:Disk instructs XCOPY to execute without actually mounting any ARCHVOLS or copying any DSNINDEX records. The result is a report indicating which DSNINDEX and ARCHVOL records would have been selected in a live run.

Because output devices are not allocated during simulations, SAMS:Disk can not accurately predict the number of output ARCHVOLS there will be in a live run. If you only want to know which DSNINDEX and/or ARCHVOL records will be selected for input by XCOPY, then this is not a concern. However, certain steps need to be taken if your intent is to determine the approximate number of ARCHVOLS that will be required for output.

XCOPY has a facility to simulate specific device types to predict volume capacity, and when a volume switch would occur. This is accomplished by specifying the appropriate DYNnUNIT sysparm with the SAMS:Disk value for the device type to be simulated as indicated below:

1. The MERP1TYP (and MERC1TYP, if desired) sysparm should specify one of the following Media Option types:

Media Option	Sysparm in Effect
3480	DYNCUNIT
DYN1	DYN1UNIT
DYN2	DYN2UNIT
DYN3	DYN3UNIT

2. The DYNnUNIT sysparm, related to the above Media Options, should be specified with one of the following cartridge device types:

Table 8-2. List of Cartridge Devices Simulated

Value	Device Simulated	Capacity
01600	3420 at 1600 BPI	
06250	3420 at 6250 BPI	
34800	6485	200 MB

XCOPY Commands and Parameters

The Disaster Recovery Extract Utility uses three (3) commands to process the data:

**SET MODE=
SELECT
XCOPY**

SET MODE= Command

Use this command to define environment values that will be in effect for the duration of the SYSIN. You may put the SET command anywhere in the command structure, but before the first action command.

Parameters

SIMULATE

Specifies that this is a simulation run. In simulate mode, SAMS:Disk will produce the normal messages and reports as if processing had taken place in *LIVE* mode, but will not alter any data sets.

To switch the processing environment into *LIVE* mode, simply comment out the SET statement by placing an asterisk (*) in column 1, or change the value to LIVE.

LIVE

This parameter is optional and the default value for the SET MODE= command. By specifying LIVE, you are instructing SAMS:Disk to execute the XCOPY command in live mode. Data sets selected by the SELECT DSN= command will be processed according to XCOPY parameters.

Syntax

```
//SYSIN DD *  
SET MODE=parameter
```

Example

```
//SYSIN DD *  
SET MODE=LIVE
```

SELECT Command

The SELECT command indicates the archive and backup data set names to be copied from input ARCHVOLS. Any number of SELECT commands is supported.

Parameters

DSN=

A list of data set names to be selected as input to the XCOPY command. This list can be fully qualified data set names, or pattern names. Finally, this list can be optionally enclosed on a single SELECT command within parenthesis.

Syntax

```
//SYSIN DD *
SELECT DSN=(value1,value2,value3)
```

Example

```
//SYSIN DD *
SET MOD=SIMULATE
SELECT DSN=(PROD1./,PROD3./)
SELECT DSN=SYSA.*.LOADLIB
SELECT DSN=PRODACCT.MASTER.LOADLIB
SELECT DSN=CHKPOINT.UPDATE.LIB
```

XCOPY Command

The XCOPY command instructs SAMS:Disk to create a copy of selected data sets from ARCHVOLS, and store the DSNINDEX records and their associated ARCHVOL records in the secondary files data set.

Only one XCOPY command may be supplied per execution.

Parameters

TYPES=

This optional parameter may be specified to limit processing to those archive data sets residing on specified device types. You may specify TAPE, DISK, 3480, 3490, DYN1, DYN2, DYN3, or any combination. If this parameter is omitted, processing defaults to all device types.

DATE=

This optional parameter may specify that a date in an accepted SAMS:Disk format be used instead of the current date in determining which archived data sets have expired.

TAPEPULL=

This is an optional parameter. Specify TAPEPULL=GEN to generate a hardcopy list of the archive tape volumes selected for merging; that is, the input tapes that must be pulled from the tape library.

Specify TAPEPULL=USE to limit processing to those input volumes listed when the TAPEPULL=GEN parameter was specified on a previous execution. (The only additional optional parameters that are valid when processing in this mode are COPY and EXCLUDE.)

A detailed description of the TAPEPULL process can be found on page 317.

TDSNAMES=dsnamelist

This optional parameter may specify 1 to 20 archive volume data set names or patterns for which processing is to be limited.

INCLUDE=

A list of one or more archive volumes or volume patterns (or keys to disk archive data sets) to be included in the XCOPY input without regard to their eligibility for other reasons. Their presence in this list overrides the EXCLUDE parameter.

Note: To limit processing to **ONLY** those ARCHVOLS specified in this parameter, all other ARCHVOLS must be excluded by specifying the parameter EXCLUDE=.

EXCLUDE=

A list of one or more archive volumes or volume patterns (or keys to disk archive data sets) to be excluded from XCOPY processing (unless in an INCLUDE list). A maximum of 50 entries is supported.

FCREDT=

This optional parameter may be specified to limit the scan of ARCHVOL records to those whose creation date is greater than or equal to (from or since this creation date) the date specified via this value. The date value must be in an accepted SAMS:Disk format. This parameter and the FDAYSOLD= parameter are mutually exclusive. Do not specify both on the same XCOPY command or the command will be rejected.

FDAYSOLD=

This optional parameter may be specified to limit the scan of ARCHVOL records to those whose creation date is greater than or equal to the current date (or an alternate date specified in the DATE= parameter) minus the number of days specified on this parameter. The maximum value for this parameter is 9999. This parameter and the FCREDT= parameter are mutually exclusive. Do not specify both on the same XCOPY command or the command will be rejected.

CREDIT=

This optional parameter may be specified to limit the scan of ARCHVOL records to those whose creation date is less than or equal to the date specified via this value. The date value must be in an accepted SAMS:Disk format. This parameter and the DAYSOLD= parameter are mutually exclusive. Do not specify both on the same XCOPY command or the command will be rejected.

DAYSOLD=

This optional parameter may be specified to limit the scan of ARCHVOL records to those whose creation date is less than or equal to the current date (or an alternate date specified in the DATE= parm) minus the number of days specified on this parameter. The maximum value for this parameter is 9999. This parameter and the CREDIT= parameter are mutually exclusive. Do not specify both on the same XCOPY command or the command will be rejected.

LISTALL

Normally report records and messages generated by this function are produced only for the volumes actually processed. If you want to print statistics and messages for volumes and data set names not processed, specify this parameter on the XCOPY command.

COPY

The primary output data sets created by XCOPY can have duplicate copies created concurrently, often referred to as "true" duplexing. (This is controlled by the dynamic allocation SYSPARMS or by providing the proper JCL directly.) This COPY parameter indicates whether or not an additional copy is to be produced in an extra job step following the primary XCOPY.

If "true" duplexing is performed in the primary XCOPY, this parameter causes the keys of the duplicate copies to be written to a disk data set and passed to the subsequent step where they can be copied again. If "true" duplexing is not done, the key of the primary (instead of the duplicate) is passed to the copy step. This parameter merely causes the keys of the appropriate new archive volumes to be written (saved) in a disk data set. You must request procedure MERGCOPY directly after the XCOPY procedure if you want to actually execute the extra copy step. An ARCHVOLS record will be made for each new copy volume and properly chained from the primary (or duplicate) ARCHVOLS record.

FLAG

FLAG is provided to indicate that the original ARCHVOLS record in the input files data set is to be flagged as processed upon completion of processing. When used in conjunction with the CHKFLAG parameter, copy will automatically bypass volumes already processed in prior XCOPY executions. This will automate XCOPY's selection of volumes so the process may be run at any time to XCOPY volumes that have been newly created since the last XCOPY operation.

CHKFLAG

Will cause the program to bypass processing for any ARCHVOL previously flagged as having been processed by the FLAG parameter.

EXPDT=, RETPD=

EXPDT and RETPD have been added to provide the ability to set the expiration date for all data sets extracted by the XCOPY function. These parameters are mutually exclusive. Whether or not these parameters are specified, ARCHVOL expiration dates will be set according to the existing rules for SAMS:Disk MERGE processing (i.e., DYNEXPDT sysparm will be used). For details, please turn to page 139 in the *Systems Guide*.

If these output ARCHVOLS are being sent offsite and you normally use the External Data Manager for your output tape support, you may want to consider updating the XCOPY executing program name so that your output tapes are not managed by EDM. You can do this by updating your JCL or PROC MI= statement to specify a program name of MI=000 instead of MI=002. This will run the XCOPY PROC under the alias name of ADSMI000 and allow the output to have a valid expiration date instead of "permanent retention" if you set DYNEXPDT accordingly. This would be recommended if IXMAINT processing on the output files data set is being done at the offsite location (or not at all), or if the output tapes are not in your local tape library management system.

DEVTYPE=

By default, XCOPY processes all device types. This optional parameter may be used to limit processing to only those DSNINDEX records residing on a particular device type. Possible values are: 8598 or 6485.

Syntax

```
//SYSIN DD *
XCOPY parameter1,parameter2,parameter3, etc...
```

Example

```
//SYSIN DD *
SET MOD=LIVE
SELECT DSN=(APP1./,APP2./)
SELECT DSN=SYSA.*.LOADLIB
SELECT DSN=PRODACCT.MASTER.LOADLIB
SELECT DSN=CHKPOINT.UPDATE.LIB
XCOPY FLAG,CHKFLAG,TDSN=(APP1.BKUP/,APP2.BKUP/)
```

Sample JCL

The XCOPY command shown below selects all archive volumes that have not already been processed by XCOPY, and creates a second set of archive and backup data sets. The new backup data sets are stored in an ARCHVOL created in the secondary files data set allocated to the OFILES DD statement.

```
//COPYTAPE      EXEC XCOPY
//XCOPY.FILES   DD   DSN=DISK.PRIMARY.FILES,DISP=SHR
//XCOPY.OFILES   DD   DSN=DISK.OFFSITE.FILES,DISP=SHR
//XCOPY.SYSPARMS DD   *
MERPLNAMDMS.MERGPRI1
MERPL1TYP3480
MERC1NAMDMS.MERGCOP1
MERC1TYPDISK
//XCOPY.SYSIN    DD   *
SELECT DSN=(PROD1./,PROD3./)
SELECT DSN=SYSA./
SELECT DSN=PRODACCT.MASTER.LOADLIB
SELECT DSN=CHKPOINT.UPDATE.LIB
XCOPY FLAG,CHKFLAG,TDSN=(APP1.BKUP/,APP2.BKUP/)
```

Figure 8-1. Sample JCL for XCOPY Command

1995.195 JUL 14, 1995		X C O P Y D A T A S E T S T A T U S							PAGE 1					
FRIDAY 10.03 AM		SAMS:DISK 8.2.0												
		-----OLD-----				KILO	BLOCK	ACTION	-----NEW-----				OUTPUT	
DATA SET NAME		ARCDATE	KEY	FILE	BLOCK	BYTES	COUNT	EXPDATE	TAKEN	KEY	FILE	BLOCK	DD	
-----		-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	
***** THIS IS A SIMULATION RUN														
USER1.LOADCL		1994.279	@99992	13	76	59	3	1994.309	COPIED	PRM001	1	1	1	
USER1.MESDOC		1995.054	@99970	1	1	81	4	1995.084	COPIED	PRM001	7	24	1	
USER1.MESMERGE		1995.054	@99969	1	1	81	4	1995.084	COPIED	PRM001	6	20	1	
USER1.SRCHFOR.LIST		1995.054	@99968	1	1	81	4	1995.084	COPIED	PRM001	5	16	1	
USER1.SYSDOC		1995.054	@99967	1	1	81	4	1995.084	COPIED	PRM001	4	12	1	
USER1.SYSDOC.FINAL		1995.054	@99966	1	1	81	4	1995.084	COPIED	PRM001	2	4	1	
USER1.SYSDOC.UPDATE		1995.055	@99959	1	1	81	4	1995.085	COPIED	PRM001	3	8	1	
USER1.SYSMERGE		1995.053	@99978	1	1	81	4	1995.083	COPIED	PRM001	15	56	1	
USER1.USEDOC		1995.053	@99977	1	1	81	4	1995.083	COPIED	PRM001	14	52	1	
USER1.USEDOC.FINAL		1995.053	@99976	1	1	81	4	1995.083	COPIED	PRM001	13	48	1	
USER1.USEDOC.UPDATE		1995.053	@99975	1	1	81	4	1995.083	COPIED	PRM001	12	44	1	
USER1.USEMERGE		1995.053	@99974	1	1	81	4	1995.083	COPIED	PRM001	11	40	1	
USER1.UTIL.CNTL		1995.053	@99973	1	1	81	4	1995.083	COPIED	PRM001	10	36	1	
USER1.VAM412.PARMLIB		1995.053	@99972	1	1	81	4	1995.083	COPIED	PRM001	9	32	1	
USER1.VSMTST.LOADLIB		1995.053	@99971	1	1	81	4	1995.083	COPIED	PRM001	8	28	1	
1995.195 JUL 14, 1995		X C O P Y V O L U M E S T A T U S							PAGE 1					
FRIDAY 10.03 AM		SAMS:DISK 8.2.0												
		-----ORIGINAL-----							-----COPIED-----					
ARCHIV	VOL			DATA	BLOCK	KILO	NO			DATA	BLOCK	KILO	NO	
KEY	CNT	DATA SET NAME	EXPDATE	SETS	COUNT	BYTES	FEET	ACTION	TAKEN	SETS	COUNT	BYTES	FEET	
-----		-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	
***** THIS IS A SIMULATION RUN														
@99959	1	DISK.RHH.PXCOPY.DMSC.D1995191.T164253	1999.365	2	7	141	3	COPIED		2	7	141	3	
@99962	1	DISK.RHHTTEST.DMSC.D1995054.T114954	1999.365	1	4	80	2	COPIED		1	4	80	2	
@99963	1	DISK.RHHTTEST.DMSC.D1995054.T112641	1999.365	1	4	80	2	COPIED		1	4	80	2	
@99964	1	DISK.RHHTTEST.DMSC.D1995054.T103547	1999.365	1	4	80	2	COPIED		1	4	80	2	
@99965	1	DISK.RHHTTEST.DMSC.D1995054.T103036	1999.365	1	4	80	2	COPIED		1	4	80	2	
@99969	1	DISK.RHHTTEST.DMSC.D1995054.T095203	1999.365	1	4	80	2	COPIED		1	4	80	2	
@99970	1	DISK.RHHTTEST.DMSC.D1995054.T094755	1999.365	1	4	80	2	COPIED		1	4	80	2	
@99971	1	DISK.RHHTTEST.DMSC.D1995053.T164637	1999.365	1	4	80	2	COPIED		1	4	80	2	
@99972	1	DISK.RHHTTEST.DMSC.D1995053.T163844	1999.365	1	4	80	2	COPIED		1	4	80	2	
@99973	1	DISK.RHHTTEST.DMSC.D1995053.T163556	1999.365	1	4	80	2	COPIED		1	4	80	2	
@99974	1	DISK.RHHTTEST.DMSC.D1995053.T163352	1999.365	1	4	80	2	COPIED		1	4	80	2	
@99975	1	DISK.RHHTTEST.DMSC.D1995053.T161840	1999.365	1	4	80	2	COPIED		1	4	80	2	
@99976	1	DISK.RHHTTEST.DMSC.D1995053.T161126	1999.365	1	4									

Figure 8-2. Sample OUTPUT From the XCOPY Command

User Exits

Specifics regarding the use of user exits can be found in the User Exits section of the *Systems Guide*. Only a brief description is presented here.

XCPARCEXmmmmmmmmmm

This module is called during XCOPY processing to allow selection processing based on criteria other than that available through SELECT and XCOPY parameters. For details, please turn to page 271 in the *Systems Guide*.

XCPDSNEXmmmmmmmmmm

This module is called during XCOPY processing to allow selection processing based on criteria other than that available through SELECT and XCOPY parameters. For details, please turn to page 272 in the *Systems Guide*.

Record Formats

There are 3 Records Formats related to XCOPY:

1. RPTNDSNX — described on page 376 of the *Systems Guide*.
2. RPTNVOLX — described on page 378 of the *Systems Guide*.
3. ARCHVOLS Subfile — described on page 419 of the *Systems Guide*.

Chapter 9. Files Data Set Maintenance

The functions in this section provide convenient means to manage the index entries for archived data sets and the volumes that contain them.

Command Summary: Program Executed:

LISTV	ADSDM483
LISTD	ADSDM483
LISTF	ADSDM483
DSNDELETE	ADSDM485
VOLDELETE	ADSDM485
PURGE	ADSDM485
REBUILD	ADSDM240
DELETE	ADSDM179
CHANGE	ADSDM179
RESET	ADSDM179
BUILDARC	ADSDM179

Procedures available:

- IXMAINT
- LISTD
- REBUILD
- IXUPDATE

General Information

The archival and backup functions assist you in managing your online storage, but the copies in the archives (and the index entries for them) must also be managed. This section describes the utilities and procedures that provide you with the ability to do so.

SAMS:Disk combines the data sets being archived (or backed up) into a single output data set. This output data set is in sequential format, and therefore may be placed on a standard tape, a tape cartridge, or any disk device. Since archive and backup functions have traditionally directed their output to tape, the output data set has also corresponded one for one to an output volume. This has led us to think of them as “archive volumes”, with SAMS:Disk keeping track of them in an “archvols” index. But keep in mind that each is really just a sequential data set that may be on either disk or tape.

Whenever a data set needs to be restored, rapid access into the “archvols data set” is desired, regardless of whether it is on disk or tape. To provide this ability, SAMS:Disk also maintains an index to each of the archived data sets. The index entry contains the exact location of each archived data set within the single “archvols data set”.

In the following discussion, any reference to an archive volume refers to an “archvols data set”, which may reside on either disk or tape, and may consist of backup and/or archive copies of numerous data sets.

Each data set in the SAMS:Disk archives has its own expiration date (assigned by the requestor or an installation default), and SAMS:Disk is designed to manage the archives by examining this date for each data set. Since the expiration date for each data set is the controlling factor, SAMS:Disk default processing assigns all of the archive volumes (tapes or disk data sets) the “never expire” date of 99365. When SAMS:Disk determines that all of the data sets on an archive volume have expired, the archive volume will be deleted/expired as well.

When a data set’s expiration date is less than or equal to the current date, the data set is considered expired. Consequently, an archvols data set may contain expired as well as unexpired data sets. When an archvol’s expiration date has passed, all of the data sets contained within it should also have expired. At that time, or shortly thereafter, all of the corresponding entries should be removed from the SAMS:Disk archive index. If the archvols data set was on tape, the tape should be put back into the “scratch pool”. If it was a disk data set, it should be deleted.

Given this situation, the following items describe the most common tasks dealing with files data maintenance:

1. The most common task may be to simply list the archive volumes and/or the names of the archived data sets.

2. Perhaps the most important task is to delete index records for expired data sets from the DSNINDEX subfile. This process both frees up space in the subfile and releases archive volumes when all data sets they contain have expired.
3. Combine two or more partially used archvols into more fully used “archvols”. A partially used archvol may consist of a tape that had very little data written to it, or it may be a tape or disk archvol data set that contains many expired entries. The unexpired entries are combined onto a new archvol data set that is more fully used, freeing up either disk space or tapes to be returned to the scratch pool. Detailed information regarding this facility is contained in the MERGE section, beginning on page 313 of this manual.
4. Delete data sets from the archives regardless of their expiration status. Normally this is done on an as-requested basis to satisfy a specific need (that is, delete obsolete data sets from the archives).
5. Delete archvols from the archives, regardless of their expiration status. This may be done when a tape volume becomes lost or unusable. At the time the tape volume is removed from the archives, all index records pertaining to data sets that existed on the tape must also be deleted.
6. Make a copy of an archvol data set for backup purposes. SAMS:Disk records the presence of a backup archvol such that it may be used when a primary archvol becomes unusable. This may be done automatically by certain functions, such as Archive, Merge, Xcopy, or it may be done at any time by specific request.
7. Change the status of data sets by changing control information for them in the archive index. Users may want to change the expiration date of a data set in the archive index. It may need to be either extended or reduced.
8. Rebuild index entries from an existing archive data set (on either disk or tape), into the SAMS:Disk archive index. This can be useful in transporting data to another SAMS:Disk system, or to recover archive index data in the event it is lost.

The following chart indicates the SAMS:Disk function that can be used to perform the appropriate files data set maintenance function:

Table 9-1. Files Data Set Maintenance Commands

Function	PROC/PGM	Command
List the archive index entries.	LISTD	LISTD LISTV LISTF
Delete data set index entries that have expired.	IXMAINT IXMAINT	DSNDELETE VOLDELETE
Delete all index entries pertaining to expired volumes.	IXMAINT	VOLDELETE
Delete data set index entries regardless of their expiration dates.	IXMAINT IXUPDATE	PURGE DELETE
Delete all index entries pertaining to a volume, regardless of their expiration dates.	IXMAINT	PURGE
Make a copy of an archive volume.	ADSDM227	——
Recreate archive index records from data on an existing archive volume.	REBUILD	REBUILD
Change the expiration date in the index record for a data set.	IXUPDATE	CHANGE
Change the expiration date in the index record for an archive volume.	IXUPDATE	RESET
Assign a copy volume to a primary volume.	IXUPDATE	RESET
Disable an unusable archive volume.	IXUPDATE	RESET
Build a new volume index record.	IXUPDATE	BUILDARC

Listing Archive Indexes

SAMS:Disk index entries for archive volumes and archived data sets may be listed selectively or in aggregate. In either case, the following list contains the possible condition codes that may occur.

Table 9-2. LISTD Condition Codes and Descriptions

Code	Description
0	All specified entries were listed okay
4	Nothing was selected for listing
8	One or more unique entries specified were not found
8+	Errors during the list process

JCL - Index Listings

Index entries for each of the archived data sets as well as the containing archive volumes can be listed with the following JCL. Each archive volume entry is unique, but multiple entries may occur for each archived data set, one for each time the data set was archived/backed up. The most recent (current) version will be listed first with the oldest version last.

```
//INDEXES EXEC LISTD
//SYSIN DD *
```

LISTV Command

LISTV VOLUME= ,DSNAME= ,DATE=

VOLUME=

By default, all archvol entries are listed. Specify a list of one or more keys (or key prefixes) of the archvol entries you wish to see. For tape volumes, the key is simply the tape volser. For disk archvols, the key is generated by SAMS:Disk. The format of a key used for disk archiving is presented under the sysparm description for ARCVOLID on page 121 of the *Systems Guide*.

DSNAME=

This optional parameter restricts the list of volumes displayed to only those with data set names matching the list of names or pattern names specified. For example, DSN=DMS.BACKUPS/ would list only tapes starting with these names. A tape volume with DSN=DMS.ARCHIVES would be skipped. (A maximum of 20 entries is supported.)

DATE=

This optional parameter restricts the list of volumes displayed to only those created on the specified date. The date value may be entered in any accepted SAMS:Disk format.

LISTV Flags

The flag fields of the archive volume index record use individual bits. The first byte of the flag field has the following meanings:

Table 9-3. LISTV Flag Field - First Byte

Binary	Hex	Meaning for FIRST flag field
1000 0000	80	volume is disabled
0100 0000	40	volume is a copy of another volume
0010 0000	20	volume is password-protected
0001 0000	10	volume CLOSE failed
0000 1000	08	volume not closed
0000 0100	04	volume is full (next data set did not fit)
0000 0010	02	volume created by MERGE or XCOPY process
0000 0001	01	volume written at 6250 BPI

The second byte of the flag field has the following meanings:

Table 9-4. LISTV Flag Field - Second Byte

Binary	Hex	Meaning for SECOND flag field
1000 0000	80	DYN1 special allocation needed
0100 0000	40	DYN2 special allocation needed
0010 0000	20	DYN3 special allocation needed
0001 0000	10	volume-level backup
0000 1000	08	IDRC data compaction
0000 0100	04	XCOPY has processed this ARCHVOL
0000 0010	02	ARC#BYTE 6 byte count is present
0000 0001	01	ARCDDEV pseudo device type present

For example:

A204 means primary volume is disabled, is password-protected, was created by the MERGE command, and has been processed by XCOPY.

SAMS:Disk keeps track of the number of feet on a tape and the number of tracks on disk that have been used by the archiving process. When the size of the next data set to be archived will not fit in the remainder of the archvols data set, the archvols data set is closed and a new one allocated. The data set being archived is then written as the first data set in the new archvols data set. If a data set being archived is too large for the space on a single output volume, multiple output volumes will be used. In such a case, the entry for the first volume will indicate the number of volumes (in the VOL CNT FIELD), and chain the additional volumes together via their key fields.

The sequence number (SEQ NBR) is incremented for each volume in the chain.

LISTD Command

DSNINDEX records created for each archived data set may be listed out with the LISTD command. By default, this command will list out the first 90 bytes of the record. However, there are 2 ways in which you can list out a selective portion of the 256 byte records:

1. **LISTF** — This command, described on page 293, produces a Sterling Software defined report that lists out fields pertaining to the Format-1 date information of the archived data set.
2. **FIELDS=** — This parameter of the LISTD command, described on page 286, allows the user to produce a customized report listing only those fields the user wants included in the report.

SYNTAX

The syntax of the LISTD command is as follows:

```
LISTD DSNNAME= ,VOLUME= ,DATE= ,DUPLICATES ,FIELDS=
```

DSNAME=

By default, all archived data sets are listed. Specify a list of one or more data set names or patterns to restrict the display to those entries with matching names. (A maximum of 20 entries is supported.)

VOLUME=

This optional parameter may be used to restrict the listing of data set index entries to only those that are contained on the specified archive volumes. The list may consist of tape volume serials, keys generated for the archvol data sets written to disk, or prefixes for either of them. The list must always be for the primary volumes, even if they are disabled. (A maximum of 20 entries is supported.)

DATE=

This optional parameter restricts the list of data set index entries displayed to only those created on the specified date. The date value may be entered in any accepted SAMS:Disk format.

DUPLICATES

This optional parameter indicates that all versions of the specified data sets are to be listed. If this parameter is not specified, only the most recent copy of the data set will be displayed.

FIELDS=

This optional parameter contains the list of DSNINDEX record field names you want to appear on the LISTD report. The default value is *"blank"* which deactivates

the parameter, resulting in the LISTD report retaining its original format. When a list of field names is specified, they should be specified in the column sequence desired.

A maximum of 50 fully qualified field names is support, and their combined output length must be less than or equal to 132 characters.

For example, if you were to execute the following command:

```
//jobname JOB (acct,info),etc.
//LIST EXEC LISTD
//SYSIN DD *
LISTD FIELDS=(DSNAM,VOLNO,ARCDT,ATIME,FLAG3B,
              DSORG,FLGSC,FLGUC)
/*
```

the LISTD report would print out as follows:

1995.362 DEC 28, 1995	D A T A S E T N A M E I N D E X				PAGE	1
THURSDAY 11.40 PM					SAMS:DISK 8.3.0A	
	SOURCE	ARCHIVE	ARCH	FLAG 3		
DATA SET NAME	VOLUME	DATE	TIME	12345678	DSORG	SCR UNCAT
-----	-----	-----	-----	-----	---	-----
DMSOS.RLW.CNTL	DMSE01	1995.311	1611	X.....	PO	
ISPRLW1.PI160748.F1	SSL805	1995.325	1043	X.....	PS	Y
ISPRLW1.PI160748.X1	SSL805	1995.335	1125	X.....	PS	Y
ISPRLW1.VKSDS.FILE5	DMST02	1995.342	1327	X.....	AM	Y
LABS.RLW82.P611303A	DMS901	1995.334	0903	X.....	PS	Y
LABS.RLW82.P611303S	DMS901	1995.333	1528	X.....	PS	Y

Figure 9-1. Sample output of LISTD with FIELDS=

Every field contained in the DSNINDEX record is available for selection. What this means is that even though the LISTF command is still valid, its usefulness has been significantly reduced.

For additional information regarding the DSNINDEX record format, please turn to the topic *"Subfile Record Format — (DSNINDEX)"* on page 423 of the *Systems Guide*.

The following table is a list of field names available to you, in DSECT order of the DSNINDEX record. For your convenience, the logical record length of each field is listed as well:

Table 9-5. Possible Values to LISTD FIELD= Parameter

DSNINDEX Field	FIELDS= Value	LRECL	Field Description
DSNDSNAM	DSNAM	44	fully qualified data set name
	DSNAMS	28	first 28 characters of data set name

DSNINDEX Field	FIELDS= Value	LRECL	Field Description
DSNVOLNO	VOLNO	6	volser from which data set was archived
DSNATIME	ATIME	4	time on which the data set was archived
DSNARCDT	ARCDT	8	date on which the data set was archived
DSNEXPDT	EXPDT	8	expiration date for DSNINDEX record
DSNDSORG	DSORG	3	data set organization from format-1 DSCB
DSNDEVCL	DEVCL	6	device class
DSNSCALO	SCALO	4	type of allocation
DSNPRTRK	PRTRK	4	first extent allocation
DSNDRBLK	DRBLK	4	directory block count
DSNMMBRS	MMBRS	5	member count
DSNFILNO	FILNO	5	file number on the ARCHVOL
DSNBLKNO	BLKNO	7	first block number and PRV for 6485's
DSNRELBK	RELBK	7	first relative block number
DSNBYTES	BYTES	9	byte count
DSNBLKCO	BLKCO	7	block count
DSNARKEY	ARKEY	7	key to ARCHVOLS record
	FLAG1B	8	FLAG1 in binary format
DSNFLAG1	FLAG1	5	FLAG BYTE 1
DSNFLGSC	FLGSC	1	FLAG1 BIT 0 — Scratched
DSNFLGUC	FLGUC	1	FLAG1 BIT 1 — Uncataloged
DSNFLGRC	FLGRC	1	FLAG1 BIT 2 — Recat to pseudovol
DSNFLGAR	FLGAR	8	FLAG1 BIT 3 — Explicit Archive
DSNFLGEX	FLGEX	1	FLAG1 BIT 4 — Low level I/O
DSNFLGBK	FLGBK	6	FLAG1 BIT 5 — Incremental Backup
DSNFLGUV	FLGUV	7	FLAG1 BIT 6 — PS as DA

DSNINDEX Field	FIELDS= Value	LRECL	Field Description
DSNFLGVV	FLGVV	1	FLAG1 BIT 7 — From Virtual Vol
	FLAG2B	8	FLAG2 in binary format
DSNFLAG2	FLAG2	5	FLAG BYTE 2
DSNFLAGEN	FLAGEN	7	FLAG2 BIT 0 — Generic Profile
DSNFLAGRF	FLAGRF	8	FLAG2 BIT 1 — Discrete Profile
DSNFLAGAX	FLAGAX	1	FLAG2 BIT 2 — VSAM AIX
DSNFLAGCB	FLAGCB	7	FLAG2 BIT 3 — Catalog Backup
DSNFLAGIO	FLAGIO	1	FLAG2 BIT 4 — Had I/O errors
DSNFLAGPE	FLAGPE	1	FLAG2 BIT 5 — PDSE
DSNFLAGBT	FLAGBT	4	FLAG2 BIT 6 — RACF Bit
DSNFLAGET	FLAGET	1	FLAG2 BIT 7 — Track Image
	FLAG3B	8	FLAG3 in binary format
DSNFLAGN	FLAG3	5	FLAG BYTE 3
DSNBLKFG	BLKFG	5	FLAG3 BIT 0 — Large Block Count
DSNSTRIP	STRIP	1	FLAG3 BIT 1 — Extended Sequential
DSNBSAM	BSAM	4	FLAG3 BIT 2 — BASM Backup
DSNMEMLA	MEMLA	7	FLAG3 BIT 3 — Member Level Backup
DSNBYTFG	BYTFG	10	FLAG3 BIT 4 — Number Bytes Available
	FLAG4B	8	FLAG4 in binary format
DSNFLAGN	FLAG4	5	FLAG BYTE 4
	DSS	3	FLAG4 BIT 5 - DFSMSdss Dumped
	CC	7	FLAG4 BIT 6 - Concurrent Copy
DSNMSPFL	MSPFL	4	MSP flags
DSNSMSFL	SMSFL	4	SMS flags
DSNSMSDC	SMSDC	8	SMS Data Class
DSNSMSMC	SMSMC	8	SMS Management Class

DSNINDEX Field	FIELDS= Value	LRECL	Field Description
DSNSMSSC	SMSSC	8	SMS Storage Class
DSNSMSSG	SMSSG	8	SMS Storage Group
DSNDS1CD	DS1CD	8	original CREDIT from format-1 DSCB
DSNDS1MD	DS1MD	8	MODDT from format-1 DSCB
DSNDS1UD	DS1UD	8	USEDT from format-1 DSCB
DSNDS1ED	DS1ED	8	EXPDT from format-1 DSCB
DSNBLKCT	BLKCT	7	large block count (release 8.2 and above)
DSNDS1JN	DS1JN	8	JOBNAME or accounting code from format-1
DSNALSTN	ALSTN	8	userid of last update
DSNAREL#	AREL#	6	SAMS:Disk release number at time of archive
DSNRTDTE	RTDTE	8	date of last restore
DSNRTIME	RTIME	4	last restore time
DSNRTCNT	RTCNT	3	count of restores
DSNUSERF	USERF	25	User Field
DSNRATIO	IDRC	4	IDRC Compression ratio
	MSPFLB	8	MSP Flags
DSNSMSFL	SMSFLB	8	SMS Flags
DSNSEXPT	SEXPT	5	SMSFL BIT 0 — ARCDT Altered
DSNMCBYP	MCBYP	7	SMSFL BIT 4 — Bypassed Mgmt Class
DSNEXPNS	EXPNS	10	SMSFL BIT 5 — Date Flag Altered
DSNUPDAT	UPDAT	11	SMSFL BIT 6 — ARCDT

LISTD Flags

The flag fields of the data set index record use individual bits to reflect data set attributes.

The first byte of the flag field has the following meanings:

Table 9-6. LISTD Flag Field - First Byte

Binary	Hex	Meaning for FIRST flag field
1000 0000	80	data set was scratched at time of archival.
0100 0000	40	data set was uncataloged at time of archival.
0010 0000	20	data set was recataloged to SAMS:Disk pseudo-volume.
0001 0000	10	data set was archived explicitly.
0000 0000	00	data set was archived via BACKUP.
0000 1000	08	data set was archived via EXCP support.
0000 0100	04	data set was archived via the RETAIN proc.
0000 0010	02	data set is DSORG=DA RECFM=U/V, or is DSORG=PS RECFM=U/V and was backed up as DA.
0000 0001	01	VBACKUP done without complete ENQ integrity

The second byte of the flag field has the following meanings:

Table 9-7. LISTD Flag Field - Second Byte

Binary	Hex	Meaning for SECOND flag field
1000 0000	80	data set was protected by RACF when backed up or archived, but not by a discrete profile. Could have been protected by a generic profile, or the data set's high-level-qualifier is the same as the userid of the task that backed up or archived the data set. Set by Release 7.0 and up only.
0100 0000	40	SAMS:Disk has a copy of the RACF discrete profile that protected the data set when it was backed up or archived.
0010 0000	20	data set is a VSAM alternate index that was processed separately from the base cluster.
0001 0000	10	data set is an DMF/EF Type-2 catalog backup. Set by Release 7.0 and up only.
0000 0000	00	SAMS:Disk does not have a copy of the RACF discrete profile, if any, that protected the data set when it was backed up or archived. Causes could be that the data set was not protected by a discrete RACF profile, a non-default value of SAMS:Disk sysparm RACFBKUP, or the SAMS:Disk-saved profile or RACFENCD pointer to it has been lost and the SAMS:Disk RACFCHK1 utility corrected the situation.
0000 1000	08	data set was backed up with I/O errors. Set by Release 7.8 and up only.
0000 0010	02	data set was RACF-indicated when archived or backed up. Set by Release 7.0 and up only.
0000 0001	01	extent track image backup. Set by Release 7.8 and up only.

LISTD Report Field Description

The LISTD ALLOC QTY field may be first extent tracks (or cylinders) or total tracks (or cylinders), depending on the specification of sysparm ALLOCPRI. For further information, see the sysparm description for *ALLOCPRI* on page 115 in the *Systems Guide*. The LISTD report field labeled BLKS MBRS/MULTI-VOL indicates:

1. If the data set is a PDS, this field contains the number of directory blocks and the number of members in the PDS, except in those cases where the data set was archived using DFSMSdss as the data mover. In which case the literal "DSS-DUMPED" will appear in the field.
2. If the data set is a multivolume data set (not a PDS), this field contains the number of volumes from which the data set was archived.
3. Any other conditions result in this field being blank.

The LISTD report field labeled FIRST BLOCK indicates the block number (from the beginning of the archive data set) where the data set begins.

The LISTD report field labeled BLK CNT (Data Set Name Index report) or BLOCK NAME (Archive Volumes report) indicates the number of blocks used to contain the data set.

The LISTD report field labeled KILOBYTE represents the actual number of kilobytes written to the output. If compression is used, this represents the number of kilobytes used after the data set is compressed.

LISTF Command

Index records are created for each archived data set. The first 90 bytes of the 256 byte record can be listed with the LISTD command. Information gathered from the format 1 of the data set at archive time, and other additional fields may be listed with the LISTF command.

SYNTAX

The syntax of the LISTF command is as follows:

```
LISTF DSNAME=, VOLUME=, DATE=, DUPLICATES
```

DSNAME=

By default, all archived data sets are listed. Specify a list of one or more data set names or patterns to restrict the display to those entries with matching names. (A maximum of 20 entries is supported.)

VOLUME=

This optional parameter may be used to restrict the listing of data set index entries to only those that are contained on the specified archive volumes. The list may consist of tape volume serials, keys generated for the archvol data sets written to disk, or prefixes for either of them. The list must always be for the primary volumes, even if they are disabled. (A maximum of 20 entries is supported.)

DATE=

This optional parameter restricts the list of data set index entries displayed to only those created on the specified date. The date value may be entered in any accepted SAMS:Disk format.

DUPLICATES

This optional parameter indicates that all versions of the specified data sets are to be listed. If this parameter is not specified, only the most recent copy of the data set will be displayed.

IXMAINT Utility

General Information

This utility provides the capability to process the entire SAMS:Disk archive data set name index and delete specified records. It can delete both expired and unexpired index entries. Through command parameters, you can restrict it to processing the index entries for only certain specified data sets.

Used as a general maintenance tool, IXMAINT may be run on a daily basis with preset commands to perform periodic cleanup of the index files, or it may be used to accomplish specific tasks on an as-requested basis. Its main objective is to delete index entries for unwanted data sets. (If the index entry being deleted is the last copy of an archived data set, the catalog entry can also be removed. For details, see the sysparm description for *UNCATPSU* on page 178 of the *Systems Guide*.)

If during processing only one index entry is remaining and the data set does not exist online, SAMS:Disk may issue a TSS message. This message warns that only one copy of the data set is left. The message is sent to the userid that archived or backed up the data set, or, if the index entry expiration date has been updated, to the userid that last updated the index entry. The issuance of the TSS message is controlled by the sysparm IXUSRNOT, described on page 148 of the *Systems Guide*.

Note: This message will be issued for every IXMAINT command that detects the condition, even in simulate mode.

This utility deals with the DSNINDEX subfile of the FILES data set. There is interaction with the ARCHVOLS subfile so that archive tapes or archive data sets on disk will be released from the archives when there are no longer any DSNINDEX records that refer to it.

ENQUEUE RESERVE

To ensure files data set integrity when run in a multiple CPU environment, a RESERVE is done against the volume containing the files data set. This RESERVE is done for the duration of the step. For this reason, it may be desirable to schedule the run during low-activity times. Also consider placing the files data set on a volume that will not be in use by other CPUs during this activity.

Note: You can override this RESERVE by specifying sysparm *RSUPPRES*, described on page 166 of the *Systems Guide*. However, caution should be taken to ensure that a cross system enqueue package is active, especially if the files data set resides on a shared DASD.

Tape Management Support

IXMAINT supports tape management systems as follows:

- With the appropriate interface module specified as the value to sysparms TMSCTLEX and/or TAPCTLEX

- By controlling the catalog status of the ARCHVOL
- With GDG, TAPEPOOL, and EDM managed tape expiration.

For important information, review the following areas:

1. Topic *"Tapepool Considerations"* on page 43 of the *Systems Guide*.
2. Sysparm description for *ARCTNAME* on page 120 of the *Systems Guide*.
3. User exit descriptions for *TMSCTLEX* (on page 254) and *TAPCTLEX* (on page 253), both of the *Systems Guide*.

Condition Code Settings

The following list contains possible condition codes that may be issued while executing the IXMAINT Proc.

Table 9-8. IXMAINT Condition Codes and Descriptions

Code	Description
0	successful completion
4	informational warning messages have been issued
8	critical errors encountered, processing terminated

JCL - IXMAINT

Use the JCL below as an example for index maintenance commands.

```
//S1      EXEC  IXMAINT
//SYSIN   DD    *
(commands go here)
```

Deleting Expired Data Sets

The DSNDELETE command causes all expired data sets in the archives to be deleted from the DSNINDEX. If you have specified sysparm UNCATDEL with a value of Y, it also deletes all data sets in the archives that were archived to an expiration date of Julian date 1999.000 and which are no longer cataloged.

Note: If the catalog status of a data set can not be determined because a catalog is either damaged or offline, no action will be taken.

The entire index is scanned for expired data sets. If all data sets for an archive volume are deleted through processing, the ARCHVOLS record will also be deleted and the volume will be released from the archives.

This is the recommended method of operation for those users of tape management systems that do not support EDM, such as TLMS.

If it is determined that no DSNINDEX records exist for a given ARCHVOLS record, a flag bit is checked to ensure that the volume has been closed. If the volume is still flagged as OPEN, it will not be deleted unless the creation date is less than today's date minus one (that is, CREDIT TODAY-1). This check maintains integrity even if ARCHIVE and IXMAINT are running concurrently over midnight.

For related information, please review the sysparm description for *IXMCLEAN* on page 147 of the *Systems Guide*. This sysparm is used as a general cleanup tool to correct pointer errors that exist in the files data set.

DSNDELETE Command and Parameters

```
DSNDELETE SIMULATE,DSNAMES=,VOLUMES=,COPIES=,DATE=,
LIST=,SCRATCH=,DELPREF=
```

SIMULATE

This optional parameter indicates that the execution is to be simulated. Messages will be issued as if processing has taken place, but no updates to the archive index will be performed.

DSNAMES=

This optional parameter indicates that processing is to be limited to the data set names and patterns specified in the supplied list. From 1 to 20 entries may be supplied.

VOLUMES=

This optional parameter indicates that processing is to be limited to the volumes specified in the supplied list. For archive data sets on disk (rather than tape), this list consists of the SAMS:Disk generated keys rather than the actual volumes. From 1 to 40 volume serial numbers, prefixes, or patterns may be supplied.

Note: Sysparm IXMCLEAN can increase the number of volumes selected for processing. The purpose of this sysparm (described on page 147 of the *Systems Guide*) is to cleanup broken chains between ARCHVOLS and DSNINDEX records. So even you restrict the number of volumes to be selected by specifying VOLUMES= on your command, IXMCLEAN will add to this list any unchained ARCHVOL.

COPIES=

This optional parameter can be defined 1-9999 to indicate that this number of versions of each data set are to be maintained in the archives. For example, if COPIES=3 was specified, all but the three most current unexpired versions of a data set

are to be deleted from the archives, regardless of their expiration dates. (Note: GDG data set names are not considered special for this support. For the example given, the three most current versions of each DSN.GxxxxVyy-named data set will be kept.)

A version is determined by the data set name and original volume of a data set. If multiple copies of a data set are archived on the same date they are considered to be one version of the data set. Member level backups, described in topic "*Member Level Backup*" on page 185, are not counted as a copy under this process.

For additional information on other processing options, review the sysparm descriptions for *DSNDUPVL* and *IXMDUPDT* on pages 137 and 148 respectively in the *Systems Guide*.

DATE=YYDDD

This parameter may specify a date in any accepted SAMS:Disk format to be used as the current date whenever date comparisons are made.

LIST=

Normally a message is printed for each data set deleted from the archives. If you want to suppress the messages, use this parameter.

LIST=NONE will suppress all messages. ALL is the default.

SCRATCH=

This keyword parameter tells SAMS:Disk whether or not to scratch the archive data set on disk when all DSNINDEX records that point to it have been deleted and the ARCHVOLS record is being deleted as a result.

SCRATCH=YES is the default. SCRATCH=NO will keep the archive data set on disk even though there are no more DSNINDEX records and the ARCHVOLS record is being deleted.

Deleting Expired Volumes

The ARCHVOL expiration date is established based on the value set for sysparm DYNEXPDT, described in detail on page 139 of the *Systems Guide*. The recommended value for this sysparm is E99365 (never scratch), its default value. Using this recommendation ensures that an ARCHVOL record will exist in the archives, as long as it contains one or more unexpired DSNINDEX records.

For details on how to delete an ARCHVOL created under this recommendation, please turn to the topic "*Deleting Expired Data Sets*" on page 297.

The VOLDELETE command is used to delete ARCHVOLS that were created outside of this recommendation. It causes all expired archive volumes (and its contents), either on tape or on disk, to be deleted from the archives. An archive volume (ARCHVOLS record) is considered expired if its expiration date is less than or

equal to the current date. Any ARCHVOLS record not expired will not be processed. (See *EXPDSNS* parameter on page 301 for the exception to this.)

The *EXPDSNS* parameter adds a "DSNDELETE DSN=/" function to *VOLDELETE*. Any *DSNINDEX* record that has expired by *EXPDT* or catalog control will be deleted even though the *ARCHVOL* has not expired. However, if *VOLDELETE* is expiring a volume that did not have a "never expire" expiration date, all data sets on that volume will be purged without regard to *EXPDT*.

If the *ARCHVOLS* expiration date is "never expire" or "catalog control", *VOLDELETE* will provide no processing benefit over *DSNDELETE*. *DSNDELETE* is the only command needed for removal of expired *DSNINDEX* and *ARCHVOL* records from the *FILES* data set.

If it is determined that no *DSNINDEX* records exist for a given *ARCHVOLS* record, a flag is checked to ensure that the volume has been closed. If the volume is still flagged as *OPEN*, it will not be deleted unless the creation date is less than today's date minus one (that is, *CREDIT* < *TODAY*-1) . This check maintains integrity even if *ARCHIVE* and *IXMAINT* are running concurrently over midnight.

Unlike the *DSNDELETE* command, data sets that have expired may remain in the archive index if the *ARCHVOLS* record they correspond to has not yet expired. However, the user should be very cautious if the *VOLDELETE* command is used in an environment where any of the following are in place at archive time:

```
//ARCHIVE0 DD LABEL=EXPDT=96350
//ARCHIVEC DD LABEL=EXPDT=96350
//SYSIN DD *
SCAN REALVOLS
SELECT DSN=/
ARCHIVE EXPDT=97200
OR
//ARCHIVE0 DD LABEL=RETPD=60
//ARCHIVEC DD LABEL=RETPD=60
//SYSIN DD *
SCAN REALVOLS
SELECT DSN=/
ARCHIVE RETPD=365
OR
//SYSIN DD *
SCAN REALVOLS
SELECT DSN=/
ARCHIVE EXPDT=97200
//SYSPARMS DD *
DYNEXPDT=96350
OR
//SYSIN DD *
SCAN REALVOLS
SELECT DSN=/
ARCHIVE RETPD=365
```

```
//SYSPARMS DD *
DYNEXPDT60
```

In the above examples, the SAMS:Disk-calculated ARCHVOLS expiration date is overridden. In every case, the data sets are kept in the archives for a longer period than that of the ARCHVOLS they reside on.

When the VOLDELETE command is executed in the above environment, un-expired data sets will be deleted because they exist on volumes that have expired.

VOLDELETE Command and Parameters

```
VOLDELETE SIMULATE, VOLUMES=, EXPDSNS, DATE=, LIST=,
SCRATCH=, DELPROF=
```

SIMULATE

This optional parameter indicates that the execution is to be simulated. Messages will be issued as if processing has taken place, but no updates to the archive index will be performed.

VOLUMES=

This optional parameter indicates that processing is to be limited to the volumes specified in the supplied list. For archive data sets on disk (rather than tape), this list consists of the SAMS:Disk-generated keys rather than the actual volumes. From 1 to 40 volume serial numbers, prefixes, or patterns may be supplied.

EXPDSNS

This optional parameter indicates that in addition to expired volumes, all expired data sets are also to be deleted, regardless of their presence on an unexpired volume. Specifying this parameter is equivalent to running DSNDELETE VOL=/,DSN=/ before running VOLDELETE. That is, all expired DSNINDEX records encountered during processing will be deleted regardless of what ARCHVOLS record they point to. In addition, if all DSNINDEX records pointing to a given ARCHVOLS record are deleted, the ARCHVOLS records will also be deleted. This can be confusing in that if this occurs, it is possible that an ARCHVOLS record might be deleted that was not specified on the VOLDELETE command of this job.

DATE=

This parameter may specify a date in any accepted SAMS:Disk format to be used as the current date (instead of the real current date) whenever date comparisons are made.

LIST=

Normally a message is printed for each data set deleted from the archives. If you want to suppress the messages, use this parameter.

LIST=NONE will suppress all messages. ALL is the default.

SCRATCH=

This keyword parameter tells SAMS:Disk whether or not to scratch the archive data set on disk when the ARCHVOLS record is being deleted.

SCRATCH=YES is the default. SCRATCH=NO will keep the archive data set on disk even though the ARCHVOLS record is being deleted.

Purging Archive Data Sets/Volumes

Use the PURGE command to delete entries from the archive index regardless of their expiration dates. It applies to both archived data sets (DSNINDEX records) and archive volumes (ARCHVOLS records). Like the DSNDELETE command, when all data sets are deleted that exist on an archive volume, the archive volume will be released from the archives.

The PURGE command performs an uncatalog action for the tape (per sysparm UNCATARC), uncatalogs archived data sets (per sysparm UNCATPSU), and interfaces to tape management systems to release the tape (per sysparm TMSCTLEX).

Careful consideration should be done prior to doing a PURGE. If a PURGE must be done prior to a REBUILD, to prevent the above actions from occurring, specify the following sysparms in the PURGE job:

```
'UNCATARC'
'UNCATPSUX'
'TMSCTLEX'          (sysparm followed by 8 blanks)
```

If it is necessary to retrieve one (or only a few) data set(s) from an ARCHVOLS, refer to the *REBUILD* section beginning on page 303 for a method to retrieve the data sets that will not cause any undesirable side effects that PURGE can cause.

PURGE Command and Parameters

```
PURGE SIMULATE, VOLUMES=, DSNAMES=, LIST=, SCRATCH=, DELPROF=
```

SIMULATE

This optional parameter causes messages to be issued as if processing has taken place, but no updates to the archive index will be performed.

VOLUMES=

This parameter is required if the DSNAMES parameter is not specified; otherwise it is optional.

This optional parameter indicates that processing is to be limited to the volumes specified in the supplied list. The list should consist of primary volumes only, not copy volumes. Any copy volume that is listed will be ignored. Copy volumes and multivolume chains are automatically removed when their associated primary volumes are purged. For archive data sets on disk (rather than tape), this list consists

of the SAMS:Disk-generated keys rather than the actual volumes. From 1 to 40 volume serial numbers, prefixes, or patterns may be supplied.

Note: To remove a copy volume without removing the primary, use the IXUPDATE RESET command with COPY=NULL as documented on page 307 , or purge and rebuild the primary volume.

DSNAMES=

This optional parameter (required only if the VOLUMES parameter is not specified) indicates that processing is to be limited to the names specified in the supplied list. From 1 to 20 data set names, prefixes, or patterns may be entered. The names refer to DSNINDEX entries to be purged, not to ARCHVOLS entries.

LIST=

Normally a message is printed for each data set deleted from the archives. If you want to suppress the messages, use this parameter.

LIST=NONE will suppress all messages. ALL is the default.

SCRATCH=

This keyword parameter tells SAMS:Disk whether or not to scratch the archive data set on disk when the ARCHVOLS record is being deleted.

SCRATCH=NO is the default. SCRATCH=YES will scratch the archive data set from disk after the ARCHVOLS record has been deleted.

Archive Index Rebuild Utility

A utility is available that may be used to recreate index records from the archive tapes (or archive data sets on disk) that contain the archived data sets. The support makes use of the control record and format-1 DSCB that are written with the data at the time of archival.

Use of this facility would be appropriate in any of the following situations:

- Restoring data sets at a site other than that at which the data sets were archived.
- The archive index has been destroyed and no suitable backup is available.
- Files data set logging (discussed on page 21 of the *Systems Guide*), was not in place, and the files data set could not be fully recovered to a particular point in time.
- The index entries for a given volume have either expired or have been PURGED, but the tape has not been written over yet, and one or more data sets must be restored.

Special Considerations

The rebuild process inserts data set index records into the files data set in sequence based on the data set name, archive date, and archive time of the data set to which they correspond.

A rebuild is not allowed if a volume already has existing entries in either the ARCHVOLS or DSNINDEX files. In such a case, the volume must first be purged and the rebuild resubmitted.

If it is necessary to retrieve one (or only a few) data set(s) from a tape, rather than doing a PURGE on the tape, which can cause several undesirable side effects discussed in section *"Purging Archive Data Sets/Volumes"* on page 302, consider doing the following:

1. Allocate a small temporary FILES data set
2. Perform a REBUILD on the tape pointing to the temporary FILES data set allocated in step one
3. RESTORE the data set(s) needed, again pointing to the temporary FILES data set
4. When steps 1-3 are complete, delete the temporary FILES data set

To avoid having DSNINDEX records point to copy archives, do not rebuild from archives that are considered copy volumes. Rebuild the primary archive, then copy with the COPY utility.

JCL - Rebuild Index

The ARCHVOLS and DSNINDEX records for data sets on an archive tape, or in an archive data set on disk, can be rebuilt by use of the following JCL.

```
//REBUILD EXEC REBUILD,DSN=ddd,VOL=vvv,UNIT=uuu
//SYSIN DD *
REBUILD command and parameters are entered here
```

REBUILD Command and Parameters

A REBUILD command is not needed unless you want to specify an expiration date to the ARCHVOLS index entry that is different from the default computation. See the EXPDT parameter below. All other needed information is obtained from the DD statement for the input archive tape, or the input archive data set on disk. Symbolic parameters are provided in the JCL procedure to allow you to identify the input data set. If the archive data set spans multiple volumes, the volume serial numbers must be properly specified in the JCL.

```
REBUILD EXPDT=
OR
REBUILD UNIT=
```

EXPDT=

The default expiration date for the archive volume will be the highest date found on all of the individual archived data sets. This ensures that the volume is not dropped due to either a DELETE command or a MERGE PROCESS until every data set has indeed expired. This default calculation can, however, be overridden by supplying a date in an accepted SAMS:Disk format on this parameter. This may be desirable in cases where archive retention periods are set by an installation, and not by the data set expiration date or unit.

Note: The create TIME for the ARCHVOLS record will be x'0000' after REBUILD. The exact time is not available.

UNIT=

The default unit type is determined by the UCB device type. You can override the default for DYN unit types by specifying the same esoteric name as used in the JCL. This ensures the proper ARCFLAG2 setting in the ARCHVOL record.

IXUPDATE Utility

The following set of commands has been supplied to provide a facility to update various fields in the SAMS:Disk archive data set name index (DSNINDEX), and archive volume index (ARCHVOLS). The functions of each are described prior to each command.

JCL - Index Updates

Use the following JCL to execute the volume and data set index update commands:

```
//IXUPDATE EXEC IXUPDATE
//SYSIN DD *
Any combination of DELETE,CHANGE,RESET,BUILDARC commands
/*
```

Condition Codes

The following table lists the possible condition codes that may occur while executing the IXUPDATE Proc.

Table 9-9. IXUPDATE Condition Codes and Descriptions

Code	Description
0	Successful
4	Informational messages issued
8	Errors occurred and explanatory messages issued
16	Error in the value specified for the TIMEDATE parameter

DELETE Data Set Index Command and Parameters

The DELETE command is provided to eliminate one or all of the data set index records for a specific data set.

```
DELETE DSNAME=,TIMEDATE=,ALL
```

DSNAME=

The explicit name of a data set optionally suffixed with a version number to be deleted from the DSNINDEX. (If more than one version number of a data set is being deleted at a time, delete them in “reverse” order (that is, (-3), (-2), (-1) rather than (-1), (-2), (-3)) to prevent the shifting of the relative version number as each is deleted.)

TIMEDATE=

This optional parameter may be used to provide the specific time and date of the DSNINDEX entry to be deleted. The format is HHMMYYDDD or HHMMYYYYYDDD.

ALL

The presence of this optional parameter causes all DSNINDEX records for all versions of the data set to be deleted.

CHANGE Data Set Index Command and Parameters

The CHANGE command is provided to reset the SAMS:Disk expiration date for a specific data set.

```
CHANGE DSNAME= , EXPDT= , TIMEDATE=
```

DSNAME=

The explicit name of a data set, optionally suffixed by a version number, for which the SAMS:Disk expiration date is to be changed.

EXPDT=

The date in an accepted SAMS:Disk format that is to be the new SAMS:Disk expiration date for this archived data set.

TIMEDATE=

This optional parameter may be used to provide the specific time and date of the DSNINDEX entry to be changed. The format is HHMMYYDDD or HHMMYYYYYDDD.

RESET Command and Parameters

The RESET command can be used to modify the relationship between a primary archive volume and its backup (copy) volume, or to reset the volume's expiration date.

```
RESET KEY= , COPY= , DISABLE , ENABLE , SCRATCH , EXPDT=
```

KEY=

Specify the key of the archive tape, or the key to the archive data set on disk, for which the status is to be changed.

COPY=

The serial number of the first volumes on which a backup copy of the archive data set exists. If the word NULL is specified, spaces will be moved into the backup copy field in the ARCHVOLS record being updated.

DISABLE

Specification of this parameter causes an archive tape volume (or archive data set on disk) to be disabled. This removes it from being eligible for use by SAMS:Disk restore or recover processing. All subsequent restores that would have been directed to this volume will be directed to the copy volume, if one exists.

ENABLE

This optional parameter reverses the status set by the **DISABLE** parameter. It removes the disabled status from an archive volume. The volume will be made available again to restore and recover processing.

SCRATCH

Specification of this optional parameter will result in the deletion of the archive volume record, and its associated copies, if any. It will not cause any data set index records to be deleted. This is most commonly used to delete copy volumes from the SAMS:Disk archives.

EXPDT=

This optional parameter provides the expiration date in an accepted SAMS:Disk format that is to be assigned to the ARCHVOLS record being updated.

BUILDARC Command and Parameters

The BUILDARC command is provided to allow volume records to be created by the user. In rare cases, this may be needed to compensate for the loss of a primary or backup archive data set.

If the status or backup volume serial for the volume being created does not apply, the RESET command may be used to reflect the desired values after the new record is created.

SYNTAX

The syntax of the BUILDARC command is as follows:

```
BUILDARC DSNAME= , VOLUME= , TYPE= , ARCTYPE= , EXPDT=
```

DSNAME=

This required parameter must provide the data set name to be placed in the archvols entry.

VOLUME=

This required parameter lists the 1 to 35 volumes (either tape or disk) that contain the SAMS:Disk archive data set. As is the case with system created archive volume records, the first volume serial, if tape, becomes the record key. For disk volumes, SAMS:Disk will generate a key.

TYPE=

Specify TAPE, 6485 or DISK to indicate the type of devices you listed in the VOL= parameter.

ARCTYPE=

Specify PRIME or COPY to indicate whether the ARCHVOLS record being built is to be flagged as the primary index entry or a duplicate copy entry.

EXPDT=

The expiration date in an accepted SAMS:Disk format that is to be assigned to the new ARCHVOLS record.

Making Additional Copies of Archive Tapes

If you produce an archive tape without a duplicate copy, it is possible to create a copy after the fact. There are also other reasons why you might want to copy an archive tape. If you had a primary and duplicate copy, but the primary copy was lost or damaged or otherwise not available, you might want to copy the duplicate copy to regain the protection in having two tapes.

SAMS:Disk must keep track of multiple copies of archive tapes, so when it makes a copy of an existing archive tape, the volume serial of the new copy is recorded in the index of the volume record for which the copy is made.

If you are using the copy utility to replace a volume that is no longer available, you must also use the RESET command. Refer to the RESET command documented on page 307 for instructions on how to DISABLE the volume that is no longer available.

For example, if primary volume A has copy volume B, A → B in the archvols entry. If A is destroyed and you want to create another copy C, use B as input to the copy utility, resulting in A → B → C. To create another copy D, use C as input to the utility, such that the pointer chain from A to B to C to D is kept intact. If A is destroyed, disable it. If B is destroyed, disable it. As long as the chain is intact, disabled entries will be bypassed automatically by restore, recover and merge functions.

The volumes to be copied must already have entries in the SAMS:Disk archive tape record index (ARCHVOLS). An input stream contains records indicating the keys to the archive tapes that are to be copied. They are read from the COPYFILE DD statement. Beginning in columns 1 through 16, enter the 6-byte volume serial of the tape volume to be copied. (If you are copying an archive data set residing on disk rather than tape, enter the 6-byte key to the disk archvols record.) Only one key can be supplied per statement. Multiple statements may be entered to copy multiple volumes.

Note: Only the first volume of a multivolume archive data set should be specified in the input stream. The rest of the volumes in the sequence are automatically copied when the first volume is copied. If any one of the volumes of a multivolume archive data set needs to be copied, all have to be copied so that proper volume chaining is maintained on the volume labels.

The following JCL is presented as an example for the utility. It causes a copy volume to be created for archive tape VOLSER 123456. This utility only works for tape.

```
//COPY      EXEC  PGM=ADSMI002,PARM=ADSDM227
//STEPLIB   DD    DISP=SHR,DSN=SAMS.DISK.LOADLIB
//ABNLDUMP  DD    DUMMY
//*RCHIVER  DD    DISP=OLD,UNIT=(CART,,DEFER),VOL=SER=CPYME2
//*RCHIVES  DD    DISP=OLD,UNIT=(TAPE,,DEFER),VOL=SER=COPYME
//ARCHIVEC  DD    DISP=(NEW,KEEP,DELETE),DSN=SAMS.DISK.ARCHCOPY,
//           VOL=(,,35),
//           LABEL=EXPDT=00000,UNIT=(TAPE,,DEFER)
//MSGPRINT  DD    SYSOUT=A,DCB=BUFNO=0
//FILES     DD    DISP=SHR,DSN=SAMS.DISK.FILES
//PARMLIB   DD    DISP=SHR,DSN=SAMS.DISK.PARMLIB
//SYSPRINT  DD    SYSOUT=A
//SYSUDUMP  DD    SYSOUT=A
//COPYFILE  DD    *
123456
```

Figure 9-3. Sample JCL for Making Archive Tape Copies

Note: A more flexible way to make a partial copy of multiple ARCHVOLS is discussed in the XCOPY section beginning on page 273.

Chapter 10. MERGE

This function provides a convenient means to manage the index entries for archived data sets and the volumes that contain them.

Command Summary: Program Executed:

MERGE ADSDM128

Procedures available:

- MERGE

The MERGE function is designed to aid you in managing the SAMS:Disk archive data sets that have been put on various disk, tape, tape cartridge.

SAMS:Disk data compression, activated via the DCDATACP sysparm, has no effect on MERGE. Data is handled whether its compressed or not.

Overview

Merge processing has been designed to provide maximum flexibility in managing the archive volumes on disk, tape and other storage media, and at the same time make the best use of the available resources.

When data sets are archived, each is assigned a retention period (expiration date). By definition, then, 100 percent of the data sets are unexpired on the day they are first archived. To make the most efficient use of both disk and tape, SAMS:Disk (by default) packs all of these archived data sets into a single file (if archiving to tape), or a single data set allocated on disk.

As each day passes, however, some of these data sets may expire, creating “dead space” in the archives. Given enough time, all of the data sets will expire and the tape can be recycled or the archive data set on disk can be deleted to reclaim the space. (The IXMAINT function accomplishes this.)

In summary, each archive volume (tape or disk data set) starts out as packed full of 100 percent unexpired data sets. As the days pass, this will eventually drop to zero percent; that is, all data sets expired. The important questions are, (1) how fast does this take place and (2) can you afford to wait until it drops all the way to zero percent? In many cases, tapes and disk space will be wasted if you do. The archives need to be “consolidated” to reclaim the space.

This is the function that merge performs. Each merge run consolidates the unexpired data sets from one or more tape volumes (or archive data sets on disk) onto new archive volumes, thus freeing up tape volumes and/or disk space. This reduces the space required to store the SAMS:Disk archives, as well as eliminating disk archive data sets with a large percentage of expired data sets. After the unexpired data sets have been successfully merged forward, the input data sets on disk are deleted, and input tapes are returned to “scratch status” — disk space is reclaimed and tapes are made available for reuse. All expired SAMS:Disk index entries are dropped from the archives and unexpired entries are updated to reflect their locations on the new tape or disk archives.

Keeping some archives on disk may be desirable, because it maintains easy access to the data through auto-restore processing, without operator intervention. “Disk-to-disk” merging provides an efficient means to manage these archives, also without operator intervention.

Specifying What to Merge

Since it is very common to have many archive volumes, it may not be practical to process all of them in a single merge execution. The time and machine resources required would prohibit execution. To help you control the merge process, command parameters may be supplied to control which archive data sets are to be merged. By running the merge in simulate mode and asking for a tape pull list, you

can determine prior to the live merge run precisely what the input to the merge process will be.

The most commonly used of these command parameters is `TYPES=`, which allows you to limit the scan of archive data sets to those residing on specified device types. For example, you could specify that a merge of only the disk archive data sets be run daily, since it processes without operator intervention and makes efficient use of disk space.

The `PERCENT` parameter allows you to specify the threshold of active (unexpired) space that is to be maintained in an archive data set. When the value falls below the percent value specified, the volume is selected for merge processing. This allows archive tapes and/or archive disk data sets that are adequately used to be skipped.

The `MAXVOLS` parameter may be used to limit the merge process to a maximum number of archive volumes. Processing will terminate when the maximum volume limit is reached. A warning message will be issued to indicate that some volumes have been bypassed. The next execution of merge will process the bypassed volumes.

Merge may be further limited to processing only archive volumes whose names match those specified in the `TDSNAMES=` parameter. The `NOMERGEIN` parameter excludes archive volumes created in previous Merge or Xcopy runs from being selected as input to Merge. `INCLUDE` and `EXCLUDE` parameters further refine the input by permitting specific archive volumes to be listed and acted upon.

The `DAYSOLD`, `FDAYSOLD`, `CREDIT` and `FCREDIT` parameters all limit the input based upon the “age” of the archive volumes; that is, how long ago were they created? Parameters `DATE=` and `EXPDAYS` provide additional flexibility based upon the archive expiration dates. `EXPDAYS` excludes volumes that are about to expire (why waste processing resources if they will expire and be deleted “tomorrow”?), and `DATE=` instructs merge to pretend that “today” is the date specified, rather than the current date, which obviously affects whether merge will think that a data set or volume has expired or not.

Further information on each of these can be found beginning on page 327.

Specifying Merge Output Options

Similarly, great flexibility in controlling the creation of new archive volumes is provided through command parameters and system parameters.

By using the `LIMITS` parameter, you may specify that archive data sets are to be written to up to five new archive volumes, each containing only data sets within a given range of expiration dates.

You can also control the device type(s) to which the new archives are to be written through the dynamic allocation sysparms `MERPnTYP` and `MERPnNAM`. The most suitable archive medium for each range may also be specified; for example,

one range may be written to disk archive data sets, another to tape, and yet another to tape cartridge.

Merge also allows you to produce duplicate copies concurrently with the creation of the new archive volumes. This is accomplished through dynamic allocation by specifying sysparms `MERCnTYP` and `MERCnNAM`. This process is sometimes referred to as “true” duplexing, and is the most efficient way to produce duplicate copies. The copies may also go to different device types, much like the primary copies.

You can, however, create additional copies by executing an extra job step following the primary merge. If “true” duplexing is done as described above, this extra step will create the “triplicate” copy. If “true” duplexing is not done, this extra step will create a duplicate copy. See the `COPY` command parameter on page 328 for a further description.

Implementation Concerns

Running merge can be a time-consuming process, especially when a large number of tapes are involved. You may avoid running it for some tape volumes by maximizing tape usage at archive time and ensuring that expiration dates for data sets written to each tape are not spread over a large period of time. This allows tapes to be removed from the archives in a timely manner when all data sets on each tape have expired. Use the index maintenance function `DSNDELETE` (or possibly `VOLDELETE`) to remove expired data sets and volumes from the archive index.

The merge process performs functions similar to the index maintenance `DSNDELETE` (and `VOLDELETE`) command, in that they both delete expired data sets from the archives. However, the index maintenance functions perform at a much more efficient rate. To improve merge performance, run the `DSNDELETE` (or `VOLDELETE`) index maintenance command prior to running merge.

Note: Sysparms `ARCHPACK` and `ARCDSKBZ` are used by merge in the same manner in which archive processing uses them. The packing option should (probably) never be turned off (`ARCHPACK` specified with a value of `N`) if merge is creating tapes, since it would merge forward only one data set per tape, creating an enormous number of new archive tapes, and defeating the whole purpose of merge. `ARCDSKBZ` should be set to an efficient value common to all the output devices you will be using.

Operation Concerns

Merge is designed to be run on a scheduled basis, based on your installation’s needs. As stated previously, disk-to-disk merge may be run on a daily basis, since it requires no operator intervention and can quickly free up DASD space. For tapes, merge is usually run on a weekly or monthly basis, depending on the number of archive volumes created and the amount of space that can be recovered from them. Not all SAMS:Disk archive volumes need to be run through merge. Process only

those volumes that are wasting a large amount of space, or those archive data sets on disk that can now be moved to tape for the remainder of their retention period.

During a merge run, input 6485 cartridges use the DD statement `//ARCHIVER`, while input disk drives are referenced using the DD statement `//ARCHIVED`.

Therefore, depending on what is being merged, one tape and/or one 6485 drive will be needed for input. Since the ARCHVOLS records contain device type information and data set names, SAMS:Disk can dynamically allocate each of these input types as appropriate. (JCL may be provided for tape and 6485s, if desired.)

If the output is to disk, SAMS:Disk always allocates the disk archive data set dynamically. To allocate all primary output devices dynamically, the data set names and the device types must be supplied in sysparms `MERPnNAM` and `MERPnTYP`. Concurrent duplicate output copies can also be created and dynamically allocated by supplying proper values for sysparms `MERCnNAM` and `MERCnTYP`. If the concurrent copies are not wanted, sysparms `MERCnTYP` must be set to `NULL`. For more information about these sysparms, see their sysparm descriptions beginning on page 149 in the *Systems Guide*.

(Tape or 6485 devices can be allocated via JCL if desired by supplying `//ARCHIVE1`, `//ARCHIVE2`, DD statements for the primary output, and `//ARCHIVC1`, `//ARCHIVC2`, for the copy devices.)

You may optionally specify a system parameter that allows the computer operator to terminate merge prematurely (a graceful shutdown). A message is issued to the operator at the beginning of the merge process. If the operator replies to the message to indicate that processing is to be stopped, merge will terminate when the current input volume has completed processing. Processing will complete normally, and all reports and statistics will be generated as normal. This option permits you to start merge and run until the resources are needed for something else. Thus merge can be run in otherwise unused machine time. Specify this option through sysparm `MERREPLY`.

TAPEPULL Processing

Since some installations require that all tapes be pulled from the tape library and be provided with the job that will use them before it goes into execution, a facility to generate such a list for merge processing is provided. This facility is invoked through the `TAPEPULL` parameter.

Two options are used:

1. This option generates the list of tape volumes to be used as input. This list is referred to as a `TAPE PULL LIST`, and is obtained by running merge with the `SIMULATE` and `TAPEPULL=GEN` parameters. This will accomplish two things. It will produce the list of tape volumes that will be used as input for the `LIVE` merge run, enabling operators to pull

them from the library. It will also save the list in a disk data set, such that a subsequent LIVE run may use them as input.

Note: The tape pull list will specify only tape volumes to be used for input for the merge run. If output is directed to tape, appropriate tape volumes will be called for during the live run as they are needed.

There is a significant distinction between the printed tape pull list and the list written to disk. The printed list will not contain any keys of disk archive data sets that may have been selected for processing, whereas the list written to disk (which may be used by merge in a subsequent live run) will contain these keys. It is assumed that the printed list is used to actually allow tape volumes to be pulled and readied prior to the live run.

2. This option of the TAPEPULL parameter causes merge to process the list of volumes generated in the TAPEPULL=GEN run. This is done by running MERGE without the SIMULATE parameter and with TAPEPULL=USE specified.

It is important to note that when running merge using the TAPEPULL option, the volume selection is done during step 1 when the list is being generated. Therefore, all selection criteria desired should be specified on the command at this time. In addition to the parameters specified during TAPEPULL=GEN, the only other parameters that can be specified in step 2 (TAPEPULL=USE) are COPY and EXCLUDE.

When using this function, an additional JCL statement must be supplied in the merge JCL procedure. It may be supplied as a JCL override or be included directly in the JCL procedure. The DD statement "TAPEPULL" must point to a sequential data set with the attributes RECFM=FB, LRECL=80, BLKSIZE=4000. The data set should be allocated prior to the merge run.

Use the following DD statement as an example. It must be included for both the TAPEPULL=GEN and TAPEPULL=USE options.

```
//TAPEPULL DD DSN=SAMS.DISK.MERGE.TAPEPULL,DISP=OLD
```

Abend/Restart Considerations

During merge processing, the DSNINDEX is updated to reflect the new volume of residence as each data set is successfully copied to its new volume. If the merge step terminates abnormally, the ARCHVOLS file record for the current output will have incomplete statistical fields. These are normally updated during archive CLOSE processing. The absence of this information will not prevent the restoration of a data set from the output, nor will it cause any problems during processing of the output in a subsequent merge.

If desired, you may correct the incomplete statistical information in the ARCHVOLS record by purging it from the files data set (PURGE command), then using the REBUILD command to reconstruct the ARCHVOLS and DSNINDEX records from the archive data set, whether on tape or disk. If the archive data set resides on disk, the default of SCRATCH=NO should be allowed to prevail during purge processing so that REBUILD can take place.

Merge processing may be restarted by submitting the same JCL as used in the failing job. If the same selection tests are specified, processing will continue with the data set being processed when the failure took place.

As noted above, SAMS:Disk considers any tape it has written to be a good tape, even though an abend may have occurred during processing. If you are using a tape management system in your installation, care should be taken to see that it does not treat such tapes as scratch tapes. (Note that this is not an issue if you use the EDM option of your tape management system.) Some tape management systems (TLMS for example), may default to giving “abend tapes” a one-day retention period. If this is the case, you should either change your tape management system to not do this for SAMS:Disk tapes, or take clerical means to properly save the SAMS:Disk tapes after an abend.

Simulate Mode Considerations

By using the SIMULATE parameter on a MERGE job, SAMS:Disk instructs MERGE to execute without actually mounting any ARCHVOLS or moving any DSNINDEX records. The result is a report indicating which DSNINDEX and ARCHVOL records would have been selected in a live run.

Because output devices are not allocated during simulations, SAMS:Disk cannot accurately predict the number of output ARCHVOLS there will be in a live run. If you only want to know which DSNINDEX and/or ARCHVOL records will be selected for input by MERGE, then this is not a concern. However, certain steps need to be taken if your intent is to determine the approximate number of ARCHVOLS that will be required for output.

MERGE has a facility to simulate specific device types to predict volume capacity, and when a volume switch would occur. This is accomplished by specifying the

appropriate DYNnUNIT sysparm with the SAMS:Disk value for the device type to be simulated as indicated below:

1. The MERP1TYP (and MERC1TYP, if desired) sysparm should specify one of the following Media Option types:

Media Option	Sysparm in Effect
3480	DYNCUNIT
DYN1	DYN1UNIT
DYN2	DYN2UNIT
DYN3	DYN3UNIT

2. The DYNnUNIT sysparm, related to the above Media Options, should be specified with one of the following cartridge device types:

Table 10-1. List of Cartridge Devices Simulated

Value	Device Simulated	Capacity
01600	3420 at 1600 BPI	
06250	3420 at 6250 BPI	
34800	6485	200 MB

Processing Logic

The input to a merge process, simulated or live, is the unloaded DSNINDEX file sorted by data set file number within each archive data set (by archive key). All of the various input selection tests are applied to determine which archive volumes (tapes or disk archive data sets) are to be processed.

The ARCHVOLS record corresponding to each of the selected volumes is then examined (a flag bit is tested) to ensure that the volume has been properly closed. If it has not, it could be in current use, in which case it should not be processed by merge. This condition can occur if archive and merge jobs are running at the same time. To maintain data integrity under these conditions, even if the jobs are running concurrently over midnight, SAMS:Disk will not merge an “open volume” unless its creation date is less than today’s date minus one (that is, CREDIT < TODAY-1).

As mentioned above, the DSNINDEX records are sorted by file number (position) within each archive volume. This permits all data sets being merged to be located by “skipping forward”, whether this is a tape spin or a seek of the disk heads. Each unexpired data set is merged forward to a new archive volume and its index entry updated to reflect its new location (live mode only). Its expiration date in conjunction with the LIMITS= parameter determines which of the new output devices is used. If its expiration date is greater than the highest limit specified, it is merged to the archive data set associated with the highest limit.

Index entries for expired data sets are deleted. When all index entries for a volume have been either deleted or updated to reflect their new locations on a new archive volume, the old ARCHVOLS record is deleted as well. Thus, when any of the moved data sets are to be restored, SAMS:Disk will call for the new archvol.

The expiration date assigned to each archive volume created by merge is taken from the sysparm value for DYNEXPDT, or from the value you have specified in overriding JCL. The default value of DYNEXPDT is the “never expire” Julian date of 99365. If you specify a different non-zero value, it will be used. In an EDM environment, DYNEXPDT should be left at its default value which allows DSNINDEX and ARCHVOL records to expire correctly.

Merge processing relies on the specification of sysparms to determine the average length of archive tape volumes being used. The information is needed to determine the amount of space used on each ARCHVOL. Be sure to review sysparms *CARTFEET*, *CARTPRCT*, and *TAPEFEET*, beginning on page 125 of the *Systems Guide*, and include them in the SYSPARMS member of parmlib if needed.

Note: Candidates to be merged forward are determined solely by the data set index entry’s expiration date, and not by the volume index entry’s expiration date. This means that if the EXPDT or RETPD parameters were “hard-coded” on the //ARCHIVE0 DD statement during an archive run, it is possible to have an unexpired data set on an expired archive volume. The unexpired data set will still be called for in merge processing to ensure the integrity of merge; that is, no unexpired data set will be dropped. If the expired tape has been written over or the archive data

set on disk has been deleted, errors will result when trying to locate the data set to be merged. If “hard- coded” JCL expiration dates are being used, the VOLDELETE command of IXMAINT should be run prior to running merge to prevent the above problem from occurring.

Merge processing creates report records for all volumes and data sets processed, indicating the actions taken. They are created concurrently with the MERGE and INDEX UPDATE step, but must be resorted into the proper sequence for the report writer.

JCL - MERGE Command

The MERGE command shown with the sample JCL below selects as input to the merge process those archive volumes that are less than 70 percent used. (“Archive volumes” refers to both archive tape volumes and disk archive data sets. “Less than 70 percent used” means that the unexpired data sets occupy less than 70 percent of either (a) the length of the tape, or (b) the disk space allocated to the disk archive data set.) The unexpired data sets from these selected volumes will be moved to one of three new output units, based on expiration date. Those moved to either of the first two expiration groups will have a duplicate copy created concurrently. Those moved to the third (highest) expiration group will not have a duplicate created.

```
//TAPEMERG EXEC MERGE
//MERGE.SYSPARMS DD *
MERP1NAMDMS.MERGPRI1
MERP1TYPDISK
MERC1NAMDMS.MERGCOP1
MERC1TYPTAPE
*
MERP2NAMDMS.MERGPRI2
MERP2TYPTAPE
MERC2NAMDMS.MERGCOP2
MERC2TYPTAPE
*
MERP3NAMDMS.MERGPRI3
MERP3TYP3480
MERC3TYPNULL
//MERGE.SYSIN DD *

MERGE PERCENT=70,LIMITS=(90,180,365)
```

Figure 10-1. MERGE Command JCL

Condition Codes

The following condition codes are returned from the stepname MERGE during the MERGE process:

Table 10-2. MERGE Condition Codes and Descriptions

Code	Description
0	successful execution
1	successful execution (extra copy step needed)
4	warning messages issued
5	warning messages issued (extra copy step needed)
8	error terminated processing
9	error terminated processing (extra copy step needed)
16	critical error caused merge termination

MERGE Command and Parameters

Following is a description of the MERGE command and its associated parameters. Only one MERGE command may be supplied per execution.

SYNTAX

The syntax for the MERGE command is as follows:

```
MERGE SIMULATE, TYPES=, LIMITS=, PERCENT=, DATE=, EXPDAYS=,
      TAPEPULL=, MAXVOLS=, TDSNAMES=, NOMERGEIN, INCLUDE=,
      EXCLUDE=, FCREDIT=, FDAYSOLD=, CREDIT=, DAYSOLD=,
      LISTALL, COPY, UNCATALOG, DEVTYPE=
```

SIMULATE

This parameter causes MERGE to execute without mounting any tape volumes or moving any archived data sets. It will process the DSNINDEX and ARCHVOLS files and produce reports indicating what would have occurred in a live run.

TYPES=

This optional parameter may be specified to limit processing to those archive data sets residing on specified device types. You may specify TAPE, DISK, 6485, DYN1, DYN2, DYN3, or any combination. If this parameter is omitted, processing defaults to all device types.

LIMITS=

This optional parameter governs the distribution of data sets on the new output volumes. It specifies a list of one to five limit values of either Julian dates or retention periods in ascending sequence. The retention periods are 1- to 5-digit numbers and the Julian dates are 7-digit dates (that is, yyyyddd). Each value represents a boundary for the data sets that are to be merged to that output group. When a data set's expiration date is greater than a boundary limit, it is merged to the next output group, with the exception of the last boundary limit. If a data set's expiration date exceeds even the last boundary value, it will still be merged to the last output group. (A maximum of 5 entries is supported.)

PERCENT=

This optional parameter indicates the minimum percent of used space that is to be maintained on each archive volume. When the unexpired data sets on a volume account for less than this percentage of the total volume, the volume is selected for merge processing.

1. **PERCENT=0** — Select no archive volumes based on the PERCENT parameter. Use the INCLUDE parameter for input volume specification.

2. **PERCENT=nn** — Select the volume when its percent used drops below nn.
3. **PERCENT=100** — (default) Process all archive volumes regardless of the space used on them.

DATE=

This optional parameter may specify that a date in an accepted SAMS:Disk format be used instead of the current date in determining which archived data sets have expired.

EXPDAYS=nnnn

This parameter is provided to exclude archive volumes from the merge processing if either the archive volume will expire within nnnn days or all of the data sets on the archive volume will expire within nnnn days (where nnnn is a number of days from the current date or the date specified on the DATE parameter).

The current date plus this number of days is compared to the archive volume expiration date and the highest expiration date of all the data sets on the volume. If the date is greater than either of these expiration dates, the volume is excluded from MERGE processing.

TAPEPULL=

This is an optional parameter. See the topics *"Operation Concerns"* and *"Tapepull Processing"*, beginning on page 316 for a complete description of the TAPEPULL support, including the need for the //TAPEPULL DD statement.

Specify TAPEPULL=GEN to generate a hardcopy list of the archive tape volumes selected for merging; that is, the input tapes that must be pulled from the tape library. (This list is also saved in a disk data set and used with the TAPEPULL=USE option as described below.) Normally the SIMULATE parameter is supplied along with this parameter.

Specify TAPEPULL=USE to limit processing to those input volumes listed when the TAPEPULL=GEN parameter was specified on a previous execution. (The only additional optional parameters that are valid when processing in this mode are COPY, LIMIT, and EXCLUDE.)

MAXVOLS=xxx

This optional parameter may specify a maximum number of input volumes to be processed for this execution of MERGE. Default value for this parameter is 150.

TDSNAMES=dsnamelist

This optional parameter may specify 1 to 20 archive volume data set names or patterns for which processing is to be limited.

NOMERGEIN

This optional parameter causes archive volumes created during previous merge runs to be excluded (unless also in an INCLUDE list).

INCLUDE=

A list of one or more archive volumes or patterns (or keys to disk archive data sets) to be included in the MERGE input without regard to their eligibility for other reasons. Their presence in this list overrides even the EXCLUDE and NOMERGEIN parameters.

The volumes specified in this parameter will be processed as well as those selected as a result of the PERCENT parameter. Specify PERCENT=0 to cause only these volumes to be processed. (A maximum of 100 entries is supported.)

EXCLUDE=

A list of one or more archive volumes or volume patterns (or keys to disk archive data sets) to be excluded from MERGE processing (unless in an INCLUDE list). (A maximum of 50 entries is supported.)

FCREDT=

This optional parameter may be specified to limit the scan of ARCHVOLS records to those whose creation date is greater than or equal to (“from or since this creation date”) the date specified via this value. The date value must be in an accepted SAMS:Disk format. This parameter and the FDAYSOLD= parameter are mutually exclusive. Do not specify both on the same MERGE command or the command will be rejected.

FDAYSOLD=

This optional parameter may be specified to limit the scan of ARCHVOLS records to those whose creation date is greater than or equal to the current date (or an alternate date specified in the DATE= parameter) minus the number of days specified on this parameter. The maximum value for this parameter is 9999. This parameter and the FCREDT= parameter are mutually exclusive. Do not specify both on the same MERGE command or the command will be rejected.

CREDT=

This optional parameter may be specified to limit the scan of ARCHVOLS records to those whose creation date is less than or equal to the date specified via this value. The date value must be in an accepted SAMS:Disk format. This parameter and the DAYSOLD= parameter are mutually exclusive. Do not specify both on the same MERGE command or the command will be rejected.

DAYSOLD=

This optional parameter may be specified to limit the scan of ARCHVOLS records to those whose creation date is less than or equal to the current date (or an alternate date specified in the DATE= parm) minus the number of days specified on this parameter. The maximum value for this parameter is 9999. This parameter and the

CREDIT= parameter are mutually exclusive. Do not specify both on the same MERGE command or the command will be rejected.

LISTALL

Normally report records and messages generated by this function are produced only for the volumes actually processed. If you want to print statistics and messages for volumes not processed, specify this parameter on the MERGE command.

COPY

The primary output data sets created by merge can have duplicate copies created concurrently, often referred to as “true” duplexing. (This is controlled by the dynamic allocation sysparms or by providing the proper JCL directly.) This COPY parameter indicates whether or not an additional copy is to be produced in an extra job step following the primary merge.

If “true” duplexing is performed in the primary merge, this parameter causes the keys of the duplicate copies to be written to a disk data set and passed to the subsequent step where they can be copied again. If “true” duplexing is not done, the key of the primary (instead of the duplicate) is passed to the copy step.

This parameter merely causes the keys of the appropriate new archive volumes to be written (saved) in a disk data set. You must request procedure MERGCOPY directly after the MERGE procedure if you want to actually execute the extra copy step. An ARCHVOLS record will be made for each new copy volume and properly chained from the primary (or duplicate) ARCHVOLS record.

UNCATALOG

By default, MERGE does not change the catalog status of any data set that it merges. Specify this UNCATALOG parameter to have SAMS:Disk uncatalog data sets that are cataloged for auto-restore purposes after each data set is successfully merged.

You might specify this parameter on your disk-to-tape Merge runs if you wished to prevent auto-restore from tape, but allow it from disk.

Before specifying this parameter, see the *DSNDELETE* command beginning on page 298, and the *BACKUP* command beginning on page 184, regarding the combination of sysparm UNCATDEL and an expiration date of 1999.000.

DEVTYPE=

By default, MERGE processes all device types. This optional parameter may be used to limit processing to only those DSNINDEX records residing on a particular device type. Possible values are: 8598 or 6485.

Sample MERGE Reports

1987.337 DEC 03, 1987		MERGE DATA SET STATUS										PAGE 1		
THURSDAY 4.42 PM												SAMS:Disk 8.2		
		-----OLD-----				KILO	BLOCK	ACTION	-----NEW-----				OUTPUT	
DATA SET NAME		ARCDATE	KEY	FILE	BLOCK	BYTES	COUNT	EXPDATE	TAKEN	KEY	FILE	BLOCK	DD	
-----		-----				-----	-----	-----	-----	-----				-----
LABJ.MJB.TESTKD1		1987.318	@99999	1	1	36	2	1987.348	MERGED	000619	7	13	1	
LABJ.MJB.TESTKD2		1987.318	@99998	1	1	36	2	1987.348	MERGED	000619	6	11	1	
LABJ.MJB.TESTKD3		1987.318	@99997	1	1	36	2	1987.348	MERGED	000619	5	9	1	
LABJ.MJB.TESTKD4		1987.319	@99996	1	1	36	2	1987.349	MERGED	000619	4	7	1	
LABJ.MJB.TESTKD5		1987.319	@99995	1	1	36	2	1987.349	MERGED	000619	3	5	1	
LABJ.MJB.TESTKD6		1987.323	@99993	1	1	36	2	1987.353	MERGED	000619	1	1	1	
LABJ.MJB.TESTKD6		1987.319	@99994	1	1	36	2	1987.349	MERGED	000619	2	3	1	

1987.337 DEC 03, 1987		MERGE VOLUME STATUS										PAGE 1	
THURSDAY 4.42 PM												SAMS:Disk 8.2	
		-----ORIGINAL-----								-----UNEXPIRED-----			
ARCHIV	VOL	DATA		BLOCK	KILO	NO				DATA	BLOCK	KILO	NO
KEY	CNT	DATA SET NAME	EXPDATE	SETS	COUNT	BYTES	FEET	ACTION	TAKEN	SETS	COUNT	BYTES	FEET
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----
@99993	1	LABS.MJB.ARCDSK0.DMSC.D1987323. T103416	1999.365	1	2	36	1	MERGED		1	2	36	1
@99994	1	LABS.MJB.ARCDSK0.DMSC.D1987319. T170628	1999.365	1	2	36	1	MERGED		1	2	36	1
@99995	1	LABS.MJB.ARCDSK0.DMSC.D1987319. T164658	1999.365	1	2	36	1	MERGED		1	2	36	1
@99996	1	LABS.MJB.ARCDSK0.DMSC.D15NOV87. T164630	1987.349	1	2	36	1	MERGED		1	2	36	1
@99997	1	LABS.MJB.ARCDSK0.DMSC.D1987318. T125938	1999.365	1	2	35	1	MERGED		1	2	35	1
@99998	1	LABS.MJB.ARCDSK0.DMSC.D1987318. T125859	1999.365	1	2	35	1	MERGED		1	2	35	1
@99999	1	LABS.MJB.ARCDSK0.DMSC.D14NOV87. T125737	1987.348	1	2	35	1	MERGED		1	2	35	1
000619	1	LABS.MJB.MRGPRIM.DMSC.T1987337. T164017	1999.365					CREATED		7	14	251	4

Figure 10-2. Sample MERGE Report

Chapter 11. MOVE/COPY

The Move/Copy function provides the capability to migrate data sets directly from one DASD volume to another by using a single, common command format. This frees the user from unnecessary involvement with different data set organizations, attributes and device types. Data sets are selected through the use of DSCL selection criteria. DSCL provides a powerful and convenient means to configure all or portions of many volumes. Migration capabilities include SAMS:Disk archive to disk data sets.

Move/Copy can process both VSAM and non-VSAM data sets.

Command Summary:	Program Executed:
MOVE (Implicit/Explicit)	ADSVC300
COPY (Explicit)	ADSVC300

Special PARMLIB Member:	Special Files:
User-defined REBLOCKING values	(none)
User-defined EXEMPTION lists	

Procedure available:
DMS

General Information

The basic element of this function is the ability to migrate data sets directly from one DASD volume to another using a single common command format, freeing the user from unnecessary involvement with different data set organizations, attributes and device types. This basic element is also combined with the ability to scan VTOCs from one or more volumes and process all or subsets of the data sets, to provide a powerful and convenient means to configure all or portions of many volumes. Also supported is the migration of OS catalogs (non-VSAM) that execute at several times the speed of JSFMOVE.

ENQ/DEQ

By default, exclusive access to a data set is maintained for the duration of its use for move processing. This ensures that needed catalog and scratch actions can be completed successfully. If exclusive control cannot be obtained (other users still accessing the data set) in order to do the processing, an appropriate bypass message will be issued.

By default, shared access to a data set is maintained for the duration of its use for copy processing. If shared control cannot be obtained (another user accessing the data set exclusively) in order to do the processing, an appropriate bypass message will be issued.

Scratch/Catalog Defaults

When a data set is moved, the default action is to scratch the source data set and recatalog it to the target device if it was cataloged previously. Uncataloged data sets are left uncataloged. The CATALOGALL parameter can be used to cause these to be cataloged as well.

Target Allocation

If more than one target volume is specified, the default allocation technique is to allocate each successive data set processed to the next volume in the list. A second technique of filling each volume in the list before switching to the next volume can be selected by specifying the FILL parameter.

Standard DADSM allocation facilities are used that allow a primary space request to be split into as many as five extents before issuing the SPACE NOT AVAILABLE message. Contiguous space allocation (only one extent allowed) may be requested by specifying the CONTIG parameter on the MOVE or COPY command.

For both MOVE and COPY, no attempt is made to do exact positioning of data sets on the target volume, that is, standard DADSM allocation rules are in effect: the smallest free space extent large enough to contain the new data set is used. This means that for a new pack with the VTOC at the beginning, each data set will be

placed as close to the VTOC as possible, moving progressively down the pack. If exact placements for certain data sets are desired, they can be preallocated. SAMS:Disk will ensure that all attributes in the format-1 DSCB are properly set.

Note: When processing VSAM, the MOVE command relies on SRM for volume selection. For related information, please turn to the parameter description for *NEWVOLUMES* on page 344.

Excluding Data Set Names from Migration

Data sets that are to be exempted from move or copy processing should have their data set names or patterns placed in a member in the parmlib data set. Any member name not previously used may be defined for this purpose. To invoke the use of this member, specify its name as the value for the TABLE= parameter, of the DSCL selection criteria.

Each entry consists of one data set name or data set name pattern (for example, "SYS /" or "USA.SOURCE.LIB").

Renaming Non-VSAM Data Sets

Prior to moving or copying a non-VSAM data set, SAMS:Disk will generate a newname, and then link to JSCVSUT to issue an ALTER NEWNAME command. This allows SAMS:Disk to allocate the data set on the new volume with the original data set name, without having to uncatalog the original data set.

The altered name will consist of a maximum 4 qualifier basename taken from the original data set name, and a SAMS:Disk generated low level qualifier. As an example, while moving data set *userid.APPLE.BERRY.CARROT.DOUGH.EATS*, SAMS:Disk will issue an ALTER NEWNAME resulting in the following:

```
userid.APPLE.BERRY.CARROT.DMStttt
```

where *tttt* is the hex equivalent of the current time of day.

Under the following circumstances SAMS:Disk will not rename the original data set:

- The data set to be moved or copied is cataloged in an OS CVOL.
- A COPY command with the NEWNAME parameter has been issued.

Renaming VSAM Data Sets

Prior to moving a VSAM cluster, SAMS:Disk will dynamically link to JSCVSUT and issue an ALTER NEWNAME against the original cluster. The result is that the last character in the cluster name is increased by 2 alphabetic characters. This allows SAMS:Disk to define the cluster on the target volume with the original cluster name.

As an example, while moving cluster *userid.APPLE.CLUSTER*, SAMS:Disk will issue an ALTER NEWNAME resulting in the following:

```
userid.APPLE.CLUSTET
```

Note: For related information, please turn to the topic "*Error Recovery*" on page 349.

Messages

Messages are normally issued only for error conditions encountered or for special warnings. Data sets being bypassed due to the parameters specified and system defaults in effect are generally not mentioned. Sysparm VCBYPASS may be specified with a value of Y to request additional messages stating why a data set was skipped by the MOVE command; these messages are always printed by the COPY command.

The dd statement DIAGDTOC may also provide some helpful diagnostic messages while moving VSAM.

Restrictions

The following is a list of restrictions that apply solely to the Move/Copy function. Additional restrictions are documented under the heading "*General Restrictions*", located on page 19.

- Load libraries are not reblocked.
- Multi-volume non-VSAM data sets are not processed.
- APFLST or LNKLST data sets. SAMS:Disk will bypass unless explicitly selected by data set name.

Moving Data to Different Device Types

Whenever data sets are being moved between different device types, SAMS:Disk attempts to adjust the required space allocation accordingly, based on the BLKSIZE, RECFM, and DSORG for each data set. The accuracy of the space conversion depends on how well the actual data reflects the specified attributes.

For example, when a data set is moved from a 6587 to a 6585, an allocation adjustment to the target 6585 must be made. This adjustment is necessary because the track size of a 6587 is larger than that of a 6585. In this particular case, the allocation size of the target data set will most likely increase. In fact, it is normal for a 1 cylinder 6587 data set that is moved to a 6585 to increase to 2 cylinders on the target volume, as illustrated below:

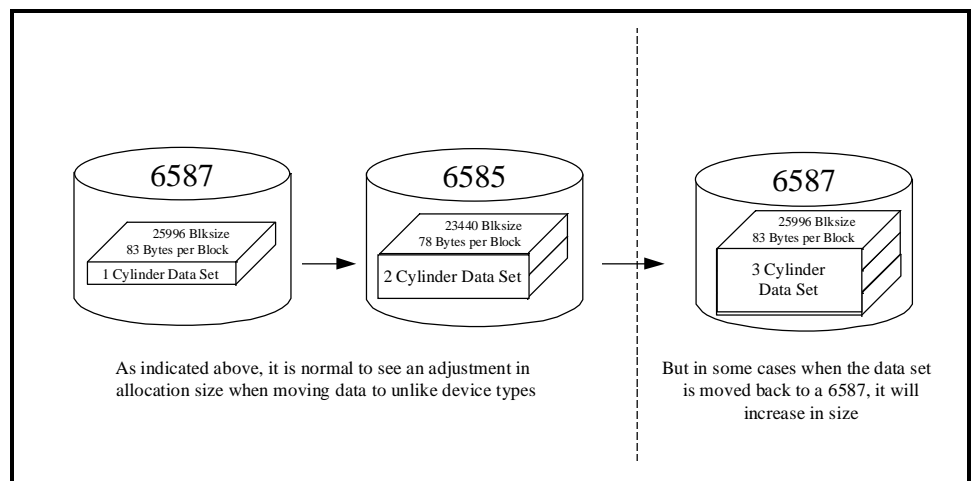


Figure 11-1. Moving a Data Set from 6587 to 6585

However, if SAMS:Disk under-allocates space during a device conversion, you may want to consider using one or more of the following sysparms:

ADPCTPDS, ADPCTOTH, ADTRACKS

For related information, please turn to the topic "*Space Allocation Algorithm*" on page 348 of the *Systems Guide*.

Special Considerations

Move/Copy processes several types of data sets, but each type is processed in a unique way. The following discussion describes these differences.

GDG Data Sets

When moving or copying GDG data sets, the following principles will be followed in regards to catalog action:

NEWNAME:

When this parameter is specified, COPY treats the target DSNAME as a new data set creation. The target data set will always be cataloged and placed in the sphere, whether the sphere is full or not. The following occurs:

1. If a catalog entry exists, the entry is left intact
2. If no catalog entry exists, the data set will be cataloged unless doing so would force another entry out of the catalog (i.e., the sphere is full)

Model DSCB processing

Data sets with zero extents allocated are most commonly acting as model DSCBs and should be treated separately from other data sets. Depending on how your DSCL selection criteria is set up, zero extent data sets may be automatically selected. To prevent selection of model DSCBs, use the DSCL EXCLUDE command with MODELDSCL criteria parameter.

Note: Sometimes model DSCBs do not have valid DSORG information associated with them and SAMS:Disk may bypass them for this reason. Consider specifying sysparm VCUDSORG with a value of Y to allow them to be moved.

BDAM Data Sets

Special considerations apply to the migration of BDAM data sets. The type of addressing used, indicated by the OPTCD field of the DSCB, dictates the migrate options available as described below.

Relative Block Addressing

No restrictions with respect to the combination of the source and target device types.

TTR Addressing

May be migrated between devices of the same type. Attempts to migrate between unlike devices may result in diagnostic messages and the data set being bypassed,

or the migration may be successful, depending on the device types and the data involved.

For related information, please review the sysparm description for *HONORTTR* on page 145 of the *Systems Guide*.

Absolute Addressing

A preallocated data set that meets the absolute address needs is required. If this condition is satisfied, the data set may be copied. To force an awareness that these special data sets are being copied, the **ABSOLUTE** and **PREALLOC** parameters must be used. Sysparm **UNMOVEVC** must also be specified.

RECFM=U OR V

BDAM data sets of this format are not preformatted with fixed-length records, and may contain either completely empty or partially empty tracks embedded within. They use either TTR or absolute CCHH addressing techniques, and may be migrated between devices of the same type. Attempts to migrate between unlike devices may result in diagnostic messages and the data set being bypassed, or the migration may be successful, depending on the device types and the data involved.

The empty or partially empty status of the contained tracks is maintained. For related information, please turn to the topic “*Direct Access and Physical Sequential Data Sets*” on page 13.

SYSCTLG Processing

SYSCTLG data sets may be migrated between like or unlike devices at several times the speed of IEHMOVE. A data set is treated as a system catalog when either the source and/or the target data set name is SYSCTLG. This permits the backing up of SYSCTLG data sets to DASD at high speed with the option of renaming the backup copy.

The capacity of the catalog may be changed significantly by preallocation of a larger SYSCTLG on the target volume. The migration support will format the additional space in the target allocation with empty catalog blocks, and update the TTR in the volume index control entry to reflect the additional capacity.

When a catalog is migrated between unlike device types, all TTR pointers in the catalog are converted. The conversion includes the validation of the TTR contained in the source catalog, and can detect either an invalid TTR or catalog entry type. These conditions cause the messages 425 and/or 426 to be issued, and the invalid block to be copied forward without modification. Although the invalid blocks are most likely disconnected, a machine comparison of the output from an IEHLIST of both the source and target catalogs can be used to validate the processing.

SAMS:Disk does not disconnect the CVOL catalog from the VSAM master catalog. It also does not reconnect the CVOL catalog to the VSAM master catalog. This must be done manually by the user with JSCVSUT utilities. For this reason the user

should always consider the status of the catalog when it is being moved by SAMS:Disk. Where are the catalog entries for the data sets being moved? If the catalog is being moved in the same run, it is possible that some data sets will lose their new catalog status because the catalog was copied to a new volume, but data sets copied subsequent to the catalog move were still being cataloged in the old copy of the catalog.

It is recommended that CVOL catalogs be moved as a separate operation, either before or after all other data sets are moved, but not within the same process. After a CVOL catalog has been moved, ensure that all of its connecting entries to the master catalog are rebuilt immediately.

To assist you in disconnecting and reconnecting a CVOL catalog that has been moved to a new volume, the following sample commands for the JSCVSUT utility are provided. Consult your Hitachi JSCVSUT manual for more specific information.

To delete high-level index pointers from your master catalog to your old CVOL (on the old volume), use commands similar to the following for each high level index:

```
DELETE (INDEX1) ALIAS
```

To define a new CVOL catalog in your master catalog:

```
DEFINE NONVSAM (NAME(SYSCTLG.V_____) DEVT( ) VOL( ))
```

To reconnect the high-level index pointers to the CVOL catalog on the new volume:

```
DEFINE ALIAS (NAME(INDEX1) RELATE(SYSCTLG.V_____) )
```

Reblocking

An additional feature of the Move/Copy function is the option to reblock sequential, partitioned, indexed sequential and direct access data sets that consist of either fixed- or variable-length records. (NOTE: RECFM=U, LOAD LIBRARIES, cannot be reblocked.) The maximum block size to be given to the migrated data sets is specified as a limit value to Move/Copy. As each data set is migrated, it is reblocked to the largest block size less than or equal to the limit value. Either a global limit may be specified to apply to all data sets of the above types, or different limits may be given for various combinations of device types, DSORG and RECFM. To define the different limit values, a list of reblocking entries must be placed in a member in the parmlib data set, each entry in the format as defined below.

When reblocking keyed data sets, the actual reblocked size may be less than the limit specified. If the new block size plus the key length will not fit on a track of a particular device type, SAMS:Disk will attempt to reduce the block size. If this is not possible, the keyed data set will not be reblocked.

As an aid when doing reblocking, sysparm NOBLKMSG is provided to request explanatory messages when a data set cannot be reblocked.

If you would like SAMS:Disk to reblock data sets that were allocated prior to DFP 3.1, you can use the DSCB Update Utility (described on page 65 of the *Installation Guide*) to turn the DS1REBLK bit on. This will allow system defined reblocking to occur.

Specifications

Reblocking entries are placed in a parmlib member of the user's choice. The parmlib member name is specified in the BLKSIZE= parameter of the COPY and MOVE commands. Each parmlib member entry has the following format.

```
'AAAAAABBBBBCCD'
```

AAAAAA

The six-character generic device name must be one of the following. This field is left-justified.

Table 11-1. Possible REBLOCK Parameter Device Names

8559	8595	8598			
------	------	------	--	--	--

BBBBB

The limit blocksize must be specified as a five- digit number with leading zeros as appropriate.

CC

Optional, but if present, must specify one of the following DSORGs:

- IS - Indexed Sequential
- PO - Partitioned
- PS - Sequential

D

Optional, but if present, must specify one of the following RECFMs:

- F - Fixed
- V - Variable

When reblocking lists are used, the only required fields are device type and limit blocksize (DSORG and RECFM are optional). Several entries may be given for each device type in order to specify different limits for various DSOrg and RECFM combinations. The first entry found that will apply to a selected data set is used. If no applicable entry is found, the data set is migrated without reblocking.

Note: Reblocking entries should always be enclosed in quotes when putting them into the parmlib member. This ensures that truncation does not occur as a consequence of embedded blanks in the device name field.

User Exits

Specifics regarding the use of user exits can be found beginning on page 195 of the *Systems Guide*. Only a brief description is presented here.

DSCLSCEXmmmmmmmmmm

This exit is given after DSCL has selected the data set for move or copy processing. The exit may allow processing to continue or stop processing for each data set. This exit is called for both VSAM and non-VSAM data sets. For complete details of this user exit, please turn to page 218 in the *Systems Guide*.

VCCOPYEXmmmmmmmmmm

VCDMIGEXmmmmmmmmmm

Each of these exits is given control just prior to the migration of a non-VSAM data set. The first exit is used by COPY command processing, the second by MOVE. All selection screening has been done by SAMS:Disk and the user's exit module is given a chance to override the migration. For complete details of this user exit, please turn to page 259 in the *Systems Guide*.

MOVE Command and Parameters

Data sets are moved by using the DSCL MOVE action command. Selection of data sets to be processed is determined by use of the DSCL selection criteria.

```
MOVE    ABSOLUTE, BLKSIZE=, CATALOGALL, CONTIG, CREATE, CYL,  
        FILL, PREALLOC, MERGE, REDUCE, TOVOL=,  
        TRACK, VATHRESH=, NEWVOLUMES
```

ABSOLUTE

This optional parameter is provided to permit the override of the default action of not copying a BDAM data set with absolute addressing. This parameter requires that the data set be preallocated on a like device, and that the PREALLOC parameter also be specified.

Applicable only to non-VSAM data sets.

BLKSIZE=

When reblocking is desired, this optional parameter must be specified in one of the two following ways:

- A 1- to 5-digit number to be used as a global limit for the new block sizes.
- A 1- to 8-character name of a member containing reblocking limit block sizes by device type, DSORG and RECFM.

When reblocking keyed data sets, the reblocked size may be less than the limit block size specified. For more information, please turn to the topic "Reblocking" on page 338.

Applicable only to non-VSAM data sets.

CATALOGALL

The presence of this parameter will cause even uncataloged source volume data sets to be cataloged to the target volume. Default processing leaves uncataloged data sets uncataloged, recataloging only those data sets that are cataloged at the start.

Applicable only to non-VSAM data sets.

CONTIG

Specify this parameter if the new data sets are to be moved to the target volume only if they can be allocated in one contiguous extent. (Default allocation uses standard DADSM allocation, which will usually result in one extent, but will allow up to five if the volume is badly fragmented.)

Applicable only to non-VSAM data sets.

CREATE

Specify this parameter if the creation date for the target data set is to be set equal to the current date. Default processing leaves the original creation date unaltered. Specifying this parameter will cause the DASD Billing support to charge for the target data set from its real creation date, not from the source data set's creation date.

Applicable only to non-VSAM data sets.

CYL

Specification of this parameter will force all allocations on the target volume(s) to be rounded to the next highest cylinder and the allocation type to be cylinder rather than track.

Applicable only to non-VSAM data sets.

FILL

Specification of this parameter will force the allocation of data sets to the first volume in the TOVOL= parameter list until it is full before any allocations are made to the next volume in the list. The default allocation technique is "round-robin".

Applicable only to non-VSAM data sets.

PREALLOC

This parameter must be entered to permit a move to preallocated data sets on a target volume. The default action is to bypass data sets having an allocation on a target volume. When this parameter is present, only one target volume may be specified.

Applicable only to non-VSAM data sets.

REDUCE

Specify this parameter when the space allocations on the target volumes should be based on amount of space used rather than amount allocated. This parameter is only applicable to PO and PS data set organizations with a non-zero secondary allocation amount. (See the TRACK parameter also.)

Applicable only to non-VSAM data sets.

TOVOL=

Specify this parameter to identify a list of 1 to 20 volume names or volume name patterns to which data sets are to be moved.

TRACK

When the REDUCE parameter is also present, this parameter causes PS and PO data sets allocated by cylinders but using less than one cylinder to have their allocations changed to tracks and the unused space released.

Applicable only to non-VSAM data sets.

VATHRESH=

This parameter indicates the percent of the volume (for VSAM) that is to be filled. Specify a value from 1 to 99 that indicates the maximum threshold of space SAMS:Disk is to use on a volume. For example, a value of 80 indicates that SAMS:Disk may fill a volume until 80% of its space is used and then will attempt to choose other volumes for new allocations.

NEWVOLUMES=

This parameter pertains to the migration of multi-volume data sets. When specified, it will indicate that none of the original volumes on which the data set resides be considered as candidate volumes when data sets are selected by volume.

If a data set is selected without any volume selection criteria, SAMS:Disk will only use the volumes specified in the TOVOL= parameter as candidate volumes.

If a data set is selected and volume selection criteria is used, SAMS:Disk will consider all volumes where the data set resides, minus the volumes it was selected for, plus the volumes specified in the TOVOL= parameter as candidate volumes.

For example, a data set resides on volumes A, B, and C. It is selected to be moved to volumes D, E, and F, and volume selection criteria is used (that is, SELECT VOL=A). With NEWVOLUMES specified, only volumes D, E, and F will be considered as candidate volumes. Without NEWVOLUMES specified, volumes B, C, D, E, and F will be considered.

MERGE

Specifying this parameter and the PREALLOC parameter will cause the pre-allocated target PDS to be merged by the source PDS.

The existing target PDS members will be overlaid by the same named source PDS members.

COPY Command and Parameters

Data sets are copied by using the DSCL COPY action command. Selection of data sets to be processed is determined by use of the DSCL FIND command.

Described below is the COPY command that may be used to copy data sets.

```
COPY  ABSOLUTE , BLKSIZE= , CATALOGALL , CONTIG , CREATE , CYL ,
      NEWNAME= , PREALLOC , MERGE , REDUCE , TOVOL= ,
      TRACK , VATHRESH= , NEWVOLUMES
```

ABSOLUTE

This optional parameter is provided to permit the override of the default action of not copying a BDAM data set with absolute addressing. This parameter requires that the data set be preallocated on a like device, and that the PREALLOC parameter also be specified.

Applicable only to non-VSAM data sets.

BLKSIZE=

When reblocking is desired, this optional parameter must be specified in one of the two following ways:

- A 1- to 5-digit number to be used as a global limit for the new block sizes.
- A 1- to 5-character name of a member containing reblocking limit block sizes by device type, DSORG and RECFM.

When reblocking keyed data sets, the reblocked size may be less than the limit block size specified. For more information, please turn to the topic "Reblocking" on page 338.

Applicable only to non-VSAM data sets.

CATALOGALL

The presence of this parameter will cause even uncataloged source volume data sets to be cataloged to the target volume. The default catalog action for the COPY command is no action.

Applicable only to non-VSAM data sets.

CONTIG

Specify this parameter if the new data sets are to be copied to the target volume only if they can be allocated in one contiguous extent. (Default allocation uses standard DADSM allocation, which will usually result in one extent, but will allow up to five if the volume is badly fragmented.)

Applicable only to non-VSAM data sets.

CREATE

Specify this parameter if the creation date for the target data set is to be set equal to the current date. Default processing leaves the original creation date unaltered. Specifying this parameter will cause the DASD Billing support to charge for the target data set from its real creation date, not from the source data set's creation date.

Applicable only to non-VSAM data sets.

CYL

Specification of this parameter will force all allocations on the target volume(s) to be rounded to the next highest cylinder and the allocation type to be cylinder rather than track.

Applicable only to non-VSAM data sets.

NEWNAME=

A different name may be given to the data set on the target volume if security checking permits it. A relative GDG name is permitted. This parameter is required for VSAM data sets.

NEWVOLUMES=

This parameter pertains to the migration of multi- volume data sets. When specified, it will indicate that none of the original volumes on which the data set resides be considered as candidate volumes when data sets are selected by volume.

If a data set is selected without any volume selection criteria, SAMS:Disk will only use the volumes specified in the TOVOL= parameter as candidate volumes.

If a data set is selected and volume selection criteria is used, SAMS:Disk will consider all volumes where the data set resides, minus the volumes it was selected for, plus the volumes specified in the TOVOL= parameter as candidate volumes.

As an example; A data set resides on volumes A, B, and C. It is selected to be copied to volumes D, E, and F, and volume selection criteria is used (that is, SELECT VOL=A). With NEWVOLUMES specified, only volumes D, E, and F will be considered as candidate volumes. Without NEWVOLUMES specified, volumes B, C, D, E, and F will be considered.

PREALLOC

This parameter must be entered to permit a copy to preallocated data sets on a target volume. The default action is to bypass data sets having an allocation on a target volume. When this parameter is present, only one target volume may be specified.

Applicable only to non-VSAM data sets.

REDUCE

Specify this parameter when the space allocations on the target volumes should be based on amount of space used rather than amount allocated. This parameter is only applicable to PO and PS data set organizations with a non-zero secondary allocation amount. (See the TRACK parameter also.)

Applicable only to non-VSAM data sets.

TOVOL=

Specify this parameter to identify a list of 1 to 20 volume names or volume name patterns to which data sets are to be moved. If this parameter is not specified, SAMS:Disk will attempt to copy the data set to its original volume. This may fail if the NEWNAME= parameter is not specified or if space does not exist on the original volume.

TRACK

When the REDUCE parameter is also present, this parameter causes PS and PO data sets allocated by cylinders but using less than one cylinder to have their allocations changed to tracks and the unused space released.

Applicable only to non-VSAM data sets.

VATHRESH=

This parameter indicates the percent of the volume (for VSAM) that is to be filled. Specify a value from 1 to 99 that indicates the maximum threshold of space SAMS:Disk is to use on a volume. For example, a value of 80 indicates that SAMS:Disk may fill a volume until 80% of its space is used and then will attempt to choose other volumes for new allocations.

MERGE

Specifying this parameter and the PREALLOC parameter will cause the pre-allocated target PDS to be merged by the source PDS.

The existing target PDS members will be overlaid by the same named source PDS members.

VSAM Cluster Considerations

SAMS:Disk processes all components of VSAM clusters selected for processing. When any VSAM component name is encountered during the scan of a volume, and selection criteria causes it to be processed, it and all of its related components will be moved or copied. Related components will be moved even if they exist on volumes other than the source volume.

Move and copy processing runs above the 16meg line for VSAM data sets.

The movement of data is accomplished by two methods, control interval method and JSCVSUT service.

Control Interval method is done via VSAM IO macros. After the data is moved, the target data set is a mirror image of the source data set. Please note that the control interval method **SHOULD NOT** be used to move the data to a different device type because the image of the target data set may not be the same as the source data set. The data is moved using the control interval method in the following cases:

1. The data set is Linear or RRDS data set.
2. The data set is ESDS and the record count fields in the statistic are zero.
3. The data set is ESDS, the record count fields in the statistic are not zero, and the value of sysparm 'VSARCFMT' is 'C'.

JSCVSUT moves data logically and any free space in the control interval is reconciled. In the event of an error, you may see JSCVSUT error messages in your SAMS:Disk report. These messages will supplement error messages issued by SAMS:Disk. The data is moved using JSCVSUT services in the following cases:

1. The data set is KSDS.
2. The data set is ESDS and the record count fields in the statistic are not zero. The sysparm 'VSARCFMT' can be used to force the data to be transferred using the control interval method.

The following special considerations apply to usage of the target volume list when processing VSAM clusters:

1. When a VSAM cluster's data and index components are moved from the same volume, they will be moved to a single target volume. When they exist on different source volumes, SAMS:Disk will attempt to place them on different target volumes.
2. Alternate index components are allocated separately from data and index components. When multiple target volumes are specified, the alternate index component may be placed on the same volume as the

base cluster, or on a different volume, depending on the results of the allocation algorithm.

3. System Resource Manager (SRM) routines select the volume to be used for the target allocation from the target volumes.
4. An attempt will be made to avoid placing cluster components on volumes that have less free space than specified on the VATHRESH parameter. If this is not possible, allocation will choose the volume with the largest free space that can contain the cluster.
5. If a data set is selected without any volume selection criteria, SAMS:Disk will only use the volumes specified in the TOVOL= parameter as candidate volumes. In other words, for a multi-volume cluster, the cluster could end up using one or more of its original volumes. To force SAMS:Disk to use only *NEW* volumes, include the NEWVOLUMES parameter described on page 344 on your command.

Error Recovery

If copy processing fails for a VSAM cluster, the target data set will be deleted and the source data set will be unchanged, as before processing started. This is an automatic process which requires no work on the part of the user.

If move processing fails for a VSAM cluster, the target data set will be deleted and the source data set will be recovered to its original status. This is an automatic process which requires no work on the part of the user.

Note: This process is done in the AMSEXEC and AMSSORT steps.

AMSSORT and AMSEXEC Steps

If SAMS:Disk abends during move processing for a VSAM cluster, the source data set is still recovered to its original state. This capability is provided by a recovery facility executed in the AMSSORT and AMSEXEC steps of the DMS procedure. These steps are only executed in the event of an abend during the VSAM move process.

Input to these two steps is provided by the AMSRFILE dd statement. The AMSEXEC step will rename any VSAM cluster left in a suspended state due to an abend based on the information provided by AMSRFILE. By default, the disposition of the AMSRFILE is a temporary data set. If a system IPL or a S222 abend occurs, you would have to manually locate the name of the source and target clusters, delete the target and alter the source back to the original name. To simplify the cleanup process, you may want to create a GDG data set to use for the AMSRFILE dd statement. This will allow the AMSRFILE to remain after an IPL or job cancel, and will provide you with the cluster name to be renamed as well as the target cluster name to be deleted during your manual cleanup.

Sample Move/Copy Commands

The following JCL shows the use of the MOVE command and the flexibility that the DSCL selection criteria provides:

```
//JOBNAME JOB (acct,info),etc.
//STEP1 EXEC DMS
SCAN REALVOLS
*****
* MOVE ALL SEQUENTIAL DATA SETS BEGINNING WITH LABS.KSV WITH *
* GREATER THAN 10 TRACKS FROM VOLUME LABS50 TO VOLUME LABS80 *
*****
SELECT DSN=LABS.KSV./,CRITERIA=(TRKS,GT,10,AND,DSORG,EQ,PS),
VOL=LABS50
MOVE TOVOL=(LABS80)
*****
* MOVE MODEL DSCBS ONLY *
*****
SELECT CRITERIA=(MODEL DSC, EQ, YES),VOL=SYS0/
MOVE TOVOL=(SYSX01)
```

Figure 11-1. Sample MOVE Command JCL

The following JCL shows the use of the COPY command:

```
//JOBNAME JOB (acct,info),etc.
//STEP1 EXEC DMS
*****
* COPY DATA SET LABS.KSV.JCL TO VOLSER DMSK01, RENAMING IT *
* TO LABS.KSV.JCL.OLD *
*****
FIND DSN=LABS.KSV.JCL
COPY TOVOL=(DMSK01),NEWNAME=LABS.KSV.JCL.OLD
```

Figure 11-2. Sample COPY Command JCL

Sample Move/Copy Report

1993.039	FEB 08, 1993	VOLUME CONFIGURATION												PAGE	1	
MONDAY	8.13 AM													SAMS:Disk	8.2	
		SOURCE DEVICE		TARGET DEVICE		DS	SRC	TRG	OP KEY		ACT	SOURCE	SOURCE	TARGET		
DATA SET NAME / NEWNAME		VOLUME	TYPE	VOLUME	TYPE	ORG	BLKSZ	BLKSZ	LRECL	CD	LEN	S	R	KBYTES	TRACKS	TRACKS
-----		-----		-----												
LABS.KSV.PARMLIB		LABS83	3380	LABS81	3380	PO	4080	4080	80	00	0	S	R	427	9	9
LABS.KSV.SOURCE		LABS83	3380	LABS81	3380	PO	4080	4080	80	00	0	S	R	332	7	7
LABS.SMP.LOADLIB		LABS83	3380	LABS81	3380	PO	6144	6144	0	00	0	S	R	1,424	30	30
LABS.KSV.MTFTABLE		LABS83	3380	LABS81	3380	PS	2994	2994	2990	00	0	S	R	94	2	2
LABS.KSV.SYSTERM		LABS83	3380	LABS81	3380	PS	4840	4840	121	00	0	S	R	94	2	2
LABS.KSV.LOADLIB		LABS83	3380	LABS81	3380	PO	6144	6144	0	00	0	S	R	189	4	4
LABS.SMP.PARMLIB		LABS83	3380	LABS81	3380	PO	4080	4080	80	00	0	S	R	94	2	2
LABS.LLM.ASMLOG		LABS83	3380	LABS81	3380	PS	132	132	132	00	0	S	R	47	1	1
LABS.LLM.LSYSPRIN		LABS83	3380	LABS81	3380	PS	4840	4840	121	00	0	S	R	47	1	1
LABS.SMP.FILES		LABS83	3380	LABS81	3380	DA	19068	19068	0	01	0	S	R	712	15	15
LABS.LLM.PARMLIB		LABS83	3380	LABS81	3380	PO	4080	4080	80	00	0	S	R	237	5	5
LABS.LLM.LOADLIB		LABS83	3380	LABS81	3380	PO	13030	13030	0	00	0	S	R	94	2	2
LABS.KSV.ASMLOG		LABS83	3380	LABS81	3380	PS	132	132	132	00	0	S	R	142	3	3
LABS.KSV.LSYSPRIN		LABS83	3380	LABS81	3380	PS	4840	4840	121	00	0	S	R	47	1	1
LABS.KSV.FILES		LABS83	3380	LABS81	3380	DA	19068	19068	0	01	0	S	R	712	15	15
LABS.SMP.ASMLOG		LABS83	3380	LABS81	3380	PS	132	132	132	00	0	S	R	47	1	1
LABS.SMP.LSYSPRIN		LABS83	3380	LABS81	3380	PS	4840	4840	121	00	0	S	R	47	1	1
* AN ASTERISK BEFORE DATA SET NAME INDICATES AN ERROR OCCURRED WHILE PROCESSING THE DATA SET																

1993.039	FEB 08, 1993	VOLUME CONFIGURATION												PAGE	2
MONDAY	8.13 AM													SAMS:Disk	8.2
		TOTAL		-----SOURCE-----				-----TARGET-----							
		DATASETS		TRACKS		KILOBYTES		TRACKS		KILOBYTES					
		-----		-----		-----		-----		-----					
MIGRATED.....		17	101	4,786		101		4,786							
SCRATCHED.....		17	101	4,786											

Figure 11-3. Sample Volume Configuration Report

Chapter 12. Idle Space Release

This function of SAMS:Disk will release unused space from partitioned and physical sequential data sets. Data sets are selected for release through the use of DSCL selection criteria. The DSCL action command RELEASE initiates the release process. The amount of unused space released may be controlled so that a portion of the unused space is retained.

Command Summary: Program Executed:

RELEASE ADSDM429

Procedure available:

- DMS

General Information

This function of SAMS:Disk will release all or part of the unused space from partitioned and physical sequential data sets. Data sets are chosen through the DSCL selection criteria and passed to the DSCL RELEASE action command for processing. Once a data set has been selected for release, the amount of space released is controlled using percent parameters.

Restrictions:

- system temporary data set names
- SYSCTLG data sets
- multivolume space allocations
- model DSCBs
- checkpoint data sets
- SAMS:Disk disk archive data sets
- all VSAM organizations

Data sets with the high level qualifier of SYS1 are bypassed by default. They may be selected through the use of sysparm RLSESYS1. Unmovable data sets and empty data sets will be selected by default using the DSCL selection criteria. To prevent space from being released from unmovable data sets, refer to the sysparm description for *RLSEUNMV* on page 165 of the *Systems Guide*, or turn to the *UNMOVE* criteria parameter on page 66.

Release processing releases idle space from empty data sets. Releasing space from empty data sets may not be desirable if it is common to allocate space for data sets in one job and then load data into them in a second job. If they are selected for processing, each will be reduced to the minimum allocation based on its type; that is, either a 1-track or a 1-cylinder data set (reduction to zero tracks allocated will not be done).

Messages

Those data sets selected for processing have their idle space released under an EXCLUSIVE ENQ (DISP=OLD) issued by SAMS:Disk. If a data set is in use by another task, a diagnostic message is issued and the release is not attempted.

The User Exit 'USERSENQ' is invalid for Idle Space Release Processing. If this User Exit is specified, an error message will be issued and a user abend will occur. Other messages are normally issued only for those cases where an attempted space release failed. If diagnostic messages stating the reasons why data sets are bypassed are also desired, sysparm RLSEDIAG must be used.

Condition Codes

For a list of condition codes, please turn to page 24.

System Parameters

The following list of sysparms are very relevant to the Idle Space Release function. For your convenience, the page number where the sysparm description can be found in the *Systems Guide* is also listed.

Table 12-1. Idle Space Release Common Sysparms

Sysparm	Page #	Description
RLSEDIAG	164	specify printing of release bypass messages
RLSEMAXS	164	specify resetting of secondary space allocation
RLSEMINs	164	specify resetting of secondary space allocation
RLSESECR	164	specify resetting of secondary space allocation
RLSESYS1	165	specify release processing for SYS1 data sets
RLSEUNMV	165	specify how to process unmovable data sets

RELEASE Command and Parameters

Release processing is done through the DSCL RELEASE action command. Selection of data sets to be processed is done by using the DSCL selection criteria.

PCTUSED=

Specify the percent of used space desired in the data sets. This value represents the percentage of used space desired in the data sets after processing. If no value is specified, 100 is the default value; that is, all space in the data set is used.

PCT SECONDARY=

Include this optional parameter to reset the amount of secondary space currently allocated for data sets selected for release of idle space. Specify a percentage (0-100) of the new primary allocation (after release of idle space) to be used to compute a new secondary space allocation.

CONVERT

This parameter causes data sets allocated by cylinders but using less than one cylinder to have their allocations changed to tracks. (The secondary space allocation quantity is also appropriately converted; that is, one 8595 cylinder converts to 30 tracks.)

Examples

The following JCL shows the use of the RELEASE command, and the flexibility that the DSCL selection criteria provides:

```

//JOBNAME JOB (acct,info),etc.
// EXEC DMS
  SCAN REALVOLS
*****
* SELECT SEQUENTIAL, NON-EMPTY DATA SETS, ALLOCATED IN      *
* TRACKS WITH SECONDARY SPACE ALLOCATION, THAT HAVE GREATER   *
* THAN 25 PERCENT USED AND LESS THAN 50 PERCENT USED.        *
*                                                              *
* RELEASE SPACE SO THAT SELECTED DATA SETS HAVE 75 PERCENT  *
* USED SPACE, AND RESET THE SECONDARY SPACE.                  *
*****
      SELECT CRITERIA=(PCTUSED,NE,0,AND,DSORG,EQ,PS,AND,
                      PCTUSED,GT,25,AND,PCTUSED,LT,50,AND,
                      ALLOTRKS,EQ,YES,AND,ALLOSEC,GE, ,
                      VOLUMES=WORK/,DSNAMES=USER./
      RELEASE PCTUSED=75,PCTSECONDARY=25
*****
* SELECT PARTITIONED DATA SETS, THAT HAVE GREATER           *
* THAN 50 PERCENT USED AND LESS THAN 90 PERCENT USED.        *
*                                                              *
* RELEASE SPACE SO THAT SELECTED DATA SETS HAVE 95 PERCENT  *
* USED SPACE,                                                 *
*                                                              *
* RESET THE SECONDARY SPACE FOR THE SELECTED PARTITIONED     *
* DATA SETS AND CONVERT CYLINDERS TO TRACKS.                 *
*****
      SELECT CRITERIA=(DSORG,EQ,PO,AND,
                      PCTUSED,GT,50,AND,PCTUSED,LT,90,AND,
                      ALLOCYLS,EQ,YES),
                      VOLUMES=MVS/,DSNAMES=TEST./
      RELEASE PCTUSED=95,PCTSECONDARY=50,CONVERT

```

Figure 12-1. Sample RELEASE Command JCL

Figure 12-2. Idle Space Release Report

Chapter 13. PDS Compression

This facility is provided to:

- Recover the dead space within a partitioned data set.
- REDUCE or INCREASE the size of the data set.
- REDUCE or INCREASE the size of the directory of the data set.
- Change the allocation type of a partitioned data set.

The reduction or increase amounts are based on the contents of the data set. PDS compression can be accomplished implicitly by the designation of volumes to be scanned for eligible data sets, or explicitly by the specification of a data set name.

Command Summary:

SCAN
COMPRESS
RELOAD

Program Executed:

ADSDM304
ADSDM304
ADSDM304

Procedure available:

- COMPRES

General Information

This facility is provided not only to recover the dead space within a PDS, but also to enlarge or shrink the size of both the data set and its directory, based upon their content. The compression is accomplished by copying each data set to a work file, calculating the reallocation requirements, then copying it back. A message is issued each time the copy-back phase of a compress is begun.

Since a single workfile is used for compressing all of the data sets, the size of the workfile must be large enough to accommodate the compressed form of the largest data set to be encountered. The workfile may be on any DASD device type (it may span multiple volumes) and given any blocksize, since SAMS:Disk restructures the data to satisfy the target workfile.

In the event that a PDS that was copied to the workfile cannot be reloaded due to an unsuccessful allocation or some other reason, **message 3179, ACTIVE WORK-FILE, CONTAINS *dsname***, will be issued and the job will terminate with a USER 200 abend. The workfile will be kept to allow the data set to be recovered via the RELOAD command. The PDS may be reloaded to the original volume or to a new volume.

PDS compression can be accomplished implicitly by the designation of volumes to be scanned for eligible data sets, or explicitly by the specification of a data set name.

The report produced by PDS Compress contains two lines for each data set processed. The first line presents the status of the data set prior to compression. The second line shows its status after compression.

Restrictions:

- APFLST or LNKLST data sets. SAMS:Disk will bypass unless explicitly selected by data set name.

Condition Codes

The following table is a list of condition codes that may occur during the execution of PDS Compress:

Table 13-1. PDS Compress Condition Codes

Code	Description
0	successful completion
4	no data sets selected or some bypassed in reallocation
8	data set(s) resource could not be acquired (for example security or enqueues)
12	unrecoverable error encountered in data set(s)
16	command error

JCL — PDS Compression

Partitioned data sets are compressed by executing the following JCL:

```
//PDSCOMP EXEC COMPRES
//SYSIN DD *
```

Enter PDS SCAN/COMPRESS commands as described below.

SCAN Command and Parameters

The SCAN command may be used to select one or several data sets for compression from one or more volumes.

During SCAN processing, data sets that are eligible for compression, but which are in use by other tasks, are noted and bypassed until the end of processing for that volume. At that time, the compress procedure re-examines the bypassed data sets to see if they are still in use by other tasks. If not, they are processed. For those data sets still in use, SAMS:Disk issues an informational message and no processing is performed.

The parameters described below specify the selection criteria as well as the reallocation values.

```
SCAN SIMULATE, VOLUMES=, EXTABLE=, DSNAME=, THRESHOLD=, EMPTY,
      USE, SHRINK, CONVERT, ALLOCATE=, RESERVE, PASSWORD
```

SIMULATE

This parameter causes the output report to be produced as if compression had been performed. However no actual compression occurs.

Note: SIMULATE has the following implications:

- Since data integrity is not an issue during SIMULATE processing, no ENQ or RESERVES are performed. This eliminates system contention for data sets and enhances performance.

Sysparm ENQDSSIM may be specified with a value of Y to cause ENQUEUEs to be issued. For further details, see the sysparm description for *ENQDSSIM* on page 141 in the *Systems Guide*.

- The directory is always read, whether you specify SIMULATE or not.
- To further enhance performance during SIMULATE processing, only the count fields of the member data blocks are read, which provides sufficient information to calculate reallocation values.

VOLUMES=

A list of one or more volumes or volume prefixes must be specified to designate the volumes to be scanned for PDS data sets to be compressed.

Default processing looks for these volumes only as online real disk packs. (A maximum of 50 entries is supported.)

EXTABLE=

Specify for this parameter the name of the parmlib member that contains entries consisting of either complete data set names or patterns that are to be excluded from compression.

This parameter can be nullified during SIMULATE runs by specifying sysparm PDSEXSIM with a value of N. This allows even the excluded data sets to be reported on during SIMULATE runs, although live runs will exclude them.

DSNAMES=

Candidates for compression may be restricted to those data sets having names or pattern names that are specified in the list value of this optional parameter. (A maximum of 10 entries is supported.)

THRESHOLD=

This parameter provides a method of limiting those data sets selected for compression. It allows you to define five conditions, one of which must be met before a PDS will be eligible for processing.

Data sets may be selected based on threshold ceilings, so you can recover dead space and/or increase allocation amounts to reduce “out of space” problems. Data sets may also be selected based on threshold floors, so you can recover overallocated space. (This parameter applies only to PDSs containing members. See the EMPTY parameter on page 364 for a similar test for PDSs containing no members.)

The conditions have the following format and meaning: (UC,XC,DC,UF,DF)

- UC “Used Ceiling” — The maximum percentage of allocated space (excluding the space allocated to the directory blocks) that is used. This value can be thought of as a “ceiling” to the threshold range. Any data set using more than this percentage of allocated space will be eligible for processing. A value of 0 will nullify this threshold test.
- XC “Extent Ceiling” — The maximum number of extents used by each data set. This can be thought of as a threshold “ceiling”. Any data set going into more than this number of extents will be eligible for processing. A value of 0 will nullify this threshold test.
- DC “Directory Ceiling” — The maximum percentage of directory space that is used. This can be thought of as a “ceiling” to the threshold range. Any data set that has a directory using more than this percentage

of allocated directory space will be eligible for processing. A value of 0 will nullify this threshold test.

- UF “Used Floor” — The minimum percentage of allocated space (excluding the space allocated to the directory blocks) that is used. This value can be thought of as a “floor” to the threshold range. Any data set using less than this percentage of allocated space will be eligible for processing. A value of 0 will nullify this threshold test.
- DF - “Directory Floor” — The minimum percentage of directory space that is used. This value can be thought of as a “floor” to the threshold range. Any data set that has a directory using less than this percentage of allocated directory space will be eligible for processing. A value of 0 will nullify this threshold test.

For conditions UC and DC, specifying a value of 100 or more will cause all data sets to fail the threshold ceiling test. For condition XC, specifying a value of 16 or more will cause all data sets to fail the threshold ceiling test.

Since these thresholds provide the upper and lower limits for data sets that are considered ineligible for processing, it is not valid to specify for conditions UF and DF a value higher than the value specified in conditions UC and DC, respectively.

See also the USE, SHRINK and CONVERT parameters for how they are used in conjunction with THRESHOLD=.

EMPTY

This parameter provides an additional method for selecting data sets for compression. By specifying EMPTY, you extend PDS Compress processing to include empty data sets (that is, PDSs containing no members).

If this parameter is not present, empty data sets will not be selected for processing. However, in explicit mode (i.e., the COMPRESS command) the SPACE= parameter will force the compress even if the data set is empty.

If the EMPTY parameter is specified along with THRESHOLD=(0,16), only empty PDSs will be selected.

Selected empty data sets are indicated by an EF in the THRESHOLD EXCEEDED field of the PDS Compression report.

When the EMPTY parameter is specified, processing is affected by the USE parameter as follows:

- If the USE parameter is present, any dead space once used for members that have been deleted is eliminated.
- If the USE parameter is not present, all primary space (excluding the space required for the directory blocks) is freed. The space thus freed

is divided by 15 (the number of possible secondary extents) and added to the current secondary allocation amount. This in effect compresses the data set down to its smallest amount while still allowing room for future expansion.

See also the **CONVERT** and **SHRINK** parameters.

When **EMPTY** is specified, the following conditions for compression may be displayed in the "THRESHOLD EXCEEDED" column of the compression report.

- **EC "Empty Ceiling"** -- This condition is triggered when the DS1LSTAR field of the Format-1 DSCB does not point to the first record after the directory. This can occur when all members have been deleted, and the data set has not yet been compressed.
- **EF "Empty Floor"** -- This condition is set for empty PDS's when **EMPTY** was specified on the command line and the **USE** parameter was not specified. The action is to free all space except that needed for the directory.

USE

The presence of this parameter forces the use of the existing data set extent(s). Any allocation changes, if specified, are confined to directory adjustment; that is, the PP and SS subparameters of the **ALLOCATE=** parameter are ignored and only the DD readjustment is done.

For empty data sets (if the **EMPTY** parameter is also specified), the presence of this parameter will cause the **ALLOCATE=** parameter to be ignored.

If the **USE** parameter is present and sysparm **PDSSENQSH** is specified with a value of **Y**, only a **SHARED ENQ** is maintained for the data set during compression. In addition, linkage editor enqueues are issued to preserve the data set's integrity.

SHRINK

Specify this parameter to allow the data set allocation to be reduced. Default processing (this parameter absent) will not allow a data set to be reduced in size. The **ALLOCATE=** parameter must be present for **SHRINK** to be active.

Empty data sets (if the **EMPTY** parameter is also specified) will be reduced in size regardless of whether or not the **SHRINK** parameter is specified. Specifying the **SHRINK** parameter, however, provides an even greater space savings, because it reduces the directory allocation down to one track (or cylinder—see **CONVERT** parameter), thus permitting the primary space allocation to also be reduced to a single track (or cylinder).

CONVERT

This parameter may be used in conjunction with the **SHRINK** parameter for data sets currently allocated in cylinders. If it is specified on the same command as the

USE parameter, CONVERT will be ignored. When a new space allocation is calculated and is less than one cylinder, this parameter converts the space allocation amount and type from cylinders to tracks (or blocks). This includes data sets allocated with the ROUND request in which the blocks are rounded up to space equal to an integral number of cylinders.

For empty data sets (if the EMPTY parameter is also specified), the CONVERT parameter will have the following effect:

- If the SHRINK parameter is not present, the primary allocation will be reduced to the minimum number of tracks required to hold the directory blocks.
- If the SHRINK parameter is also specified, the primary allocation (including the directory blocks) will be reduced to one track.

ALLOCATE=

This *optional* parameter specifies three values used to reallocate the primary space, secondary space and directory space. The format and meanings are as follows:

(PP , SS , DD)

- PP—The percentage of primary space (excluding the space allocated to the directory blocks) to be used after reallocation and compression. A value of 0 may be used to force the new allocation equal to the original.
- SS—The new secondary allocation quantity (excluding the space allocated to the directory blocks) expressed as a percentage of the primary. A value of 0, which is the default, may be used to force the new allocation equal to the original.
- DD—The percentage of directory space to be used after reallocation and compression. A value of 0, which is the default, may be used to force the new allocation equal to the original.

See the USE, SHRINK and CONVERT parameters for how they are used in conjunction with ALLOCATE=. Also see the SPACE= parameter (described under the explicit COMPRESS command below.)

RESERVE

This parameter is provided for use in a multiple- CPU environment that does not use a global enqueue facility, to protect against other CPUs making concurrent updates. It causes a RESERVE to be issued against the volume in addition to the shared or exclusive ENQ already issued on the data set name. (See USE parameter on page 365.) If this parameter is omitted, only the ENQ is done. The RESERVE and/or ENQ are done just prior to the compress of each data set and DEQ is issued immediately upon completion.

PASSWORD

This parameter is active only if sysparm **PASSWORD** has been specified with a value of Y. Specify this parameter to allow SAMS:Disk to compress password-protected data sets without prompting the operator for the password. If you do not specify this parameter, SAMS:Disk will bypass these password-protected data sets.

COMPRESS Command and Parameters

The **COMPRESS** command must be used to explicitly compress individual data sets as follows:

```
COMPRESS  SIMULATE , DSNAME= , VOLUME= , THRESHOLD= , EMPTY ,
          USE , SHRINK , CONVERT , ALLOCATE= , SPACE= , RESERVE ,
          PASSWORD
```

DSNAME=

This parameter designates the name of the PDS to be compressed.

VOLUME=

This parameter must be supplied when the PDS is not cataloged.

SPACE=

Use this parameter when you want to provide absolute values (rather than percentages) for the new space allocation of a PDS.

When processing in implicit (SCAN) mode, using percentage values for calculating the reallocation amounts permits efficient processing of numerous data sets. However, when specifying an explicitly named PDS (**COMPRESS**), the desired reallocation amounts are often known, and can be supplied directly using this parameter. **SPACE=** may be used only with explicit processing.

To request actual space reallocation quantities, specify **SPACE=** in conjunction with the **ALLOCATE=** parameter. When thus combined, the **SPACE=** parameter supplies the allocation type, while the **ALLOCATE=** parameter supplies the allocation quantities. For example, specifying

```
SPACE=TRK , ALLOCATE= ( 30 , 10 , 5 )
```

means to reallocate the PDS with 30 primary tracks and 10 secondary tracks, and a block directory.

If **SPACE=** is specified without the **ALLOCATE=** parameter, the data set will be reallocated using the new allocation type. For example, specifying

```
SPACE=CYLCTG
```

will convert the original allocation to cylinders with the **CONTIG** attribute.

If **SPACE=** is specified, the **USE**, **SHRINK**, **CONVERT**, **EMPTY**, and **THRESHOLD** parameters are ignored.

Caution: The use of the SPACE= parameter will force the compression of empty data sets, whether the EMPTY parameter is specified or not.

Specify this parameter with one of the following literal values:

Table 13-2. PDS Compress SPACE Parameter Values

Literal Value	Redefines Allocation to
CYL	cylinder request
CYLCTG	cylinder with CONTIG request
TRK	track request
TRKCTG	track with CONTIG request
BLK	avg. block length request
BLKCTG	avg. block length with CONTIG request
RND	avg. block length with ROUND request
RNDCTG	avg. block length with CONTIG and ROUND request

Note: For any type of block allocation, the blocksize is obtained from the data set's format-1 DSCB. If the blocksize is 0 (or greater than the maximum blocksize for the device), the blocksize is assumed to be the maximum for the new device; therefore the new allocation will be the same number of tracks as the number of blocks specified.

Additional Parameters

All of the following parameters have the same meanings and use as documented for the implicit SCAN command:

- SIMULATE
- EMPTY
- USE
- SHRINK
- CONVERT
- RESERVE
- PASSWORD

The following parameters can have meanings that are different from the implicit SCAN command:

THRESHOLD=

All values set to this parameter are ignored in explicit processing.

ALLOCATE=

The data set will be reallocated using the values specified for this parameter. (i.e., `ALLOCATE=(10,0,0)` will reallocate the data set with 0 secondary allocation and all unused directory blocks freed.

RELOAD Command

The RELOAD command is used to restart PDS Compress processing for a single specific data set after a system failure occurred during a PDS Compress execution. For example, if a system failure had occurred during a PDS Compress execution and the data set being compressed at the time of the failure had been scratched, reallocated, and was in the process of being reloaded, this procedure would restart the reload process to recover the PDS.

If the system failure occurred during the processing of unloading the PDS to the work data set, there is no need to use the reload command because the original data set is intact. To determine where the compress was interrupted, see the SAMS:Disk message listing. A “copy-back started” message is printed before each data set compressed message. If a “copy-back started” message is not followed by a successful compression message, the RELOAD has not completed successfully, and the RELOAD function must be run to recover the data set.

Note: A target data set for the RELOAD must be preallocated with the proper DCB attributes. The DCB attributes are printed in the “reload starting” message as a source of reference.

Use the following JCL to execute the RELOAD command. Remember, since only one PDS was left in the workfile (in the process of being compressed), only one RELOAD command is supported.

```
//S1      EXEC  COMPRES
//SYSIN   DD   *
          RELOAD  DSNAME= , VOLUME= , PASSWORD
```

DSNAME=

(Optional) If this parameter is omitted, SAMS:Disk retrieves the DSNAME from the WORKFILE DD in the COMPRES PROC. Otherwise, this is the name of the PDS that has been preallocated for recovery of the unloaded data set.

VOLUME=

(Optional) If the data set being reloaded is not cataloged, or you do not want SAMS:Disk to use the catalog to locate it, specify the DASD volume serial number on which the data set resides.

PASSWORD

(Optional) Specify this parameter to allow SAMS:Disk to reload password-protected data sets without prompting the operator for the password. If you do not

specify this parameter and the data set is password-protected, the reload will fail once again.

Sample PDS Compression Report

The PDS compression report contains two lines for each data set processed. The first line presents the status of the data set prior to compression. The second line shows its status after compression.

1987.337		DEC 03, 1987		P A R T I T I O N E D D A T A S E T C O M P R E S S I O N										PAGE		1					
THURSDAY		4.37 PM												SAMS:Disk		8.2					
THRESHOLD														ALLOCATION							
PRI		EXT		DIR												PRI		SEC		DIR	
														45		15		65			
DATA SET NAME		THRESHOLD EXCEEDED		VOLUME		DEVICE TYPE		-SPACE REQUEST- PRIME SEC		EXT		-----TRACKS----- ALLOC USED IDLE			-DIRECTORY BLOCKS- ALLOC USED IDLE			--DIRECTORY ENTRIES--- ALLOC USED IDLE LEN			
LABS.ICC.LOADLIB				LABS83 3380		BLK		35 7		1		5 2 3			10 1 9			70 3 67 36			
				LABS83 3380		BLK		49 14		1		7 2 5			10 1 9			70 3 67 36			
LABS.ICC.PARMLIB				LABS83 3380		TRK		2 1		1		2 1 1			10 2 8			60 7 53 42			
				LABS83 3380		TRK		5 1		1		5 1 4			10 2 8			60 7 53 42			
LABS.ICC.SOURCE				LABS53 3350		TRK		45 5		1		45 27 18			10 2 8			60 11 49 42			
				LABS53 3350		TRK		63 10		1		63 27 36			10 2 8			60 11 49 42			
LABS.ICC.TESTJCL				LABS81 3380		TRK		7 1		1		7 6 1			10 9 1			60 51 9 42			
				LABS81 3380		TRK		16 3		1		16 6 10			13 9 4			78 51 27 42			
LABS.ICC75.PARMLIB				WRK001 3350		TRK		28 3		1		28 24 4			20 9 11			217 93 124 22			
				WRK001 3350		TRK		56 9		1		56 24 32			13 9 4			143 93 50 22			

Figure 13-1. PDS Compress Report

Chapter 14. Sequential Migration to Tape

This SAMS:Disk facility migrates sequential data sets from disk devices to tape. The data sets are written and recataloged as standard labeled tape files, multiple files per tape. Any job that previously accessed its disk data set via the catalog will now call for a tape mount and continue running as before.

Command Summary:

MODESET
MIGRATE
SCAN
RECATALOG

Program Executed:

ADSDM205
ADSDM205
ADSDM205
ADSDM272

Special PARMLIB Members:

MIGRATXM (MET=..)
GDGDASDG

Special Files:

MIGRECAT

Procedure available:

- MIGRATE

General Information

Sequential data sets may be moved from online storage to tape and recataloged by either explicit or implicit reference. Each DASD data set migrated is written as a standard labeled tape data set and cataloged, enabling any job that formerly referenced the data set through the catalog to continue to do so without change. Exception cases are those sequential data sets that are concatenated to other DASD data sets when read. This will cause an unlike device type error at allocation time. Similarly, if all generations of a GDG are to be read in by referencing the index name only, all of the generations must reside on either DASD or tape, but not mixed.

An attempt is made to catalog all data sets written to tape. Uncataloged non-GDG data sets are cataloged, and previously cataloged data sets (correctly cataloged) are recataloged. If a GDG data set is not in the catalog, no attempt is made to catalog it because VSAM catalogs will uncatalog the oldest generation when the index is full. (See NOEMPTY parameter for the DEFINE GENERATIONDATAGROUP command in Hitachi's Access Method Services manual.)

Installation Options

Several installation sysparms are available to alter the data set selection and processing techniques. For your convenience, the page number where the sysparm description can be found in the *Systems Guide* is also listed.

Table 14-1. Sequential Migrate Installation Sysparms

Sysparm	Page #	Description
MIGYRCHG	153	may be used to cause a new tape to be mounted when the year portion of the expiration date changes. Default processing sorts all data sets to be migrated into descending sequence by expiration date, but does not cause a volume switch when the year changes.
MIGNOEXP	152	may be used to permit expired data sets to be migrated. Default processing is for unexpired data sets only.
MIGNXPDT	152	may be used to suppress the selection of data sets that have all zeros for an expiration date. Default processing treats them as unexpired and as candidates to be migrated.
MIGCATLG	150	may be used to change the catalog action for uncataloged non-GDG data sets. The default value is Y, which means to catalog all non-GDG data sets migrated to tape that were not cataloged before.

Sysparm	Page #	Description
MIGCTGDG	151	may be used to change the catalog action for uncataloged GDG data sets. The default value is N, which means no change is made in the catalog when an uncataloged GDG is migrated to tape.
MIGDSMAX	151	may be used to set an absolute limit to the number of data sets migrated to each output tape. The default is set at 9999
MIGPSDEN	152	may be used to specify tape density for simulate mode processing. It defaults to 6250. In live mode, the actual density used is properly obtained.
MIGBYPAS	150	may be used to cause messages to be printed that state why a data set was bypassed for processing.

Criteria for Selection Data Sets

Several of the available parameters are used to establish the SAMS:Disk “scan environment”, while others are used to select data sets from those being scanned. A clear distinction between the two makes it an easy task to specify parameters to achieve desired results. The distinction is summarized here, with detailed explanations beginning on page 377.

Scan Parameters

VOLUME lists	applied to online volumes
DSNAME lists	applied to GDGs and non-GDGs alike
GDGSONLY	to look at GDG data sets only
NONGDGS	to look at non-GDG data sets only

Select Parameters

MINSIZE	to select data sets by their size, with a minimum size cutoff point
MAXSIZE	to select data sets by their size, with a maximum size cutoff point
LASTUSE	to select unused data sets
GENER	to select older generations of GDG data sets

Expiration Dates Assigned to Data Sets on Tape

For every data set on tape, the header label contains the expiration date field just as the format-1 DSCB does for every data set on disk. As each data set is migrated from disk to tape, a decision is made regarding what value to supply for its expiration date on tape. Sysparms MIGEXPDT and MIGFXPDT control this decision.

The default value of N for MIGEXPDT causes each expiration date on disk (from the format-1 DSCB) to be copied forward to the tape, unless its value is all zeros. If it is zeros, SAMS:Disk will substitute the expiration date (or retention period) that you supply in the MIGRATE dd statement. If you specify a value of Y for MIGEXPDT, the JCL-supplied date will be substituted for every data set being migrated. (If a duplicate copy tape is being created, the expiration dates for it are created in the same manner but the date from the MIGRATEC dd statement is used instead, and it may be different from the primary tape.)

You should also consider that tape management systems often regard tapes created with data set expiration dates equal to the current date as being work tapes, and therefore won't log the data sets in the tape management catalog. To prevent this, sysparm MIGFXPDT is provided. Its default value of Y causes SAMS:Disk to add one day to the expiration date whenever it finds an expiration date equal to the current date.

Sequential Migration Versus Archival

Although sequential migration to tape and the archival function can both free up disk space, the following differences should also be considered. Migration to tape handles only PS data sets while archive handles all types. Data sets migrated to tape are allocated and read by the application program directly, while archived data sets require that they be restored first, although this may be done automatically by the auto-restore feature. Migrated data sets are cataloged to tape and then managed by conventional means. (This may present a problem in deciding how long to keep the tapes and how to clean up the catalog when the tapes do expire. Migrating only GDG data sets helps manage this problem considerably.) Archived data sets are managed by SAMS:Disk through the use of the index records that are kept. Index maintenance functions will recycle tapes, as well as keep the catalog clean by uncataloging appropriate data sets as they expire from the archives.

Multivolume Tape Processing

Data sets migrated to tape are written as standard labeled tape files; there are multiple files per tape. The first data set written is the dummy file created by SAMS:Disk; it is standard label file 1 (1,SL). The first data set migrated is standard label file 2 (2,SL). The file number is incremented by one for each data set on the tape. If SAMS:Disk predicts that the next data set will not fit on the current tape, it causes a new tape to be mounted and begins again with file 1.

If a data set is too large to fit on one tape (or if SAMS:Disk thought the next data set would fit but it doesn't because the tape is shorter than expected), SAMS:Disk

will continue writing the data set on additional tapes as needed. When processing for that data set is complete, SAMS:Disk will ask for a new tape to be mounted for subsequent data sets and will reset the file number to 1.

SAMS:Disk does not stack additional data sets with incremented file numbers on the end of multivolume tapes. The reason is, for the operating system to correctly locate those file numbers for subsequent jobs that need the data sets, each tape in the multivolume set must be mounted, spun to the end to find and count the file numbers, and rewind, until the correct tape and file number are found. By not allowing this, SAMS:Disk ensures that every data set needed can be found on the first tape that is mounted.

Compatibility With Tape Management Systems

SAMS:Disk uses standard system OPEN/CLOSE routines to create data sets on tape. All data that is normally passed to the system through JCL dd statements for the output tape data sets is passed to OPEN internally by SAMS:Disk. This includes the tape expiration date. This provides compatibility to any user tape management system that gets information at tape open time. SAMS:Disk is completely compatible with the way these systems control tape use.

Creation of Sequential Migrate Tape First File

The first file of each sequential migrate tape is a data set created by SAMS:Disk. This data set is a dummy file that contains no information, but its presence helps prevent data loss when you are using a tape library management system.

The expiration date of the first file is the highest expiration date on the tape. It is equal to the expiration date of the second file on the data set. When writing data sets to tape, SAMS:Disk sorts the data sets in expiration date sequence, from the highest to the lowest.

Default processing leaves this first data set uncataloged. Specify the DMYCATLG parameter of the MODESET command to catalog this data set.

The data set name SAMS:Disk assigns to this first file is a combination of the data set name from the MIGRATE JCL and a generated date and time stamp. This SAMS:Disk-generated data set name for the first file helps provide consistency between the sequential migrate tapes, and can make them easier to manage.

The base name of the data set is taken from the data set name in the MIGRATE dd statement for the primary tape, and the MIGRATEC dd statement for the copy tape. SAMS:Disk generates a 22- character appendage to the base name as follows:

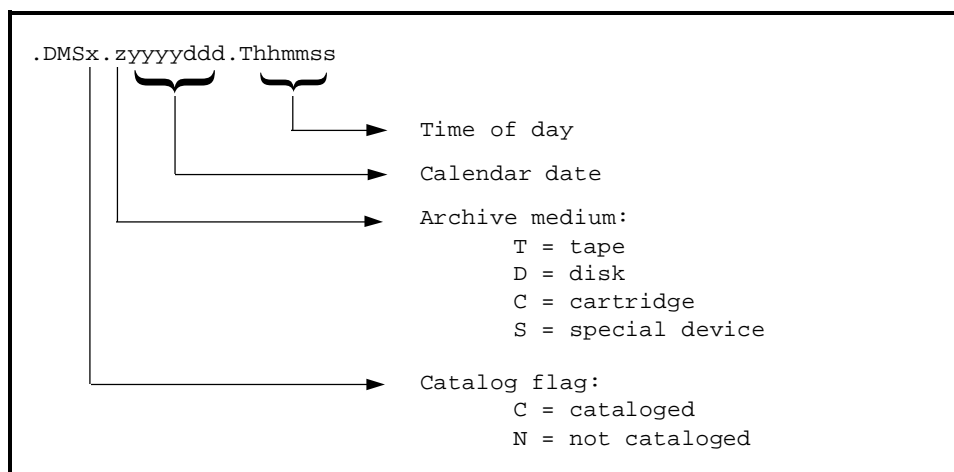


Figure 14-1. DSNNAME Generated From MIGRATEC DD

The calendar date above is the current date in Julian format, and the time of day is in hours/minutes/seconds. Therefore the first file's data set name will reflect the date and time it was created, and will always be unique. An example is SYS.MIGRATED.DMSC.C198827.T113015. When considering a base name make sure that there is enough room for the SAMS:Disk generated appendage.

Assignment of Creation Dates to Output Data Sets

As a default, SAMS:Disk resets the creation date of data sets moved to tape to the current date. This is done because the data set is truly a new data set moved to tape and because some tape management systems rely on the creation date to be equal to the current date. If you desire that SAMS:Disk leave the creation date on tape as it was found on DASD, specify sysparm MIGCREDIT with a value of Y.

Migrating Protected Data Sets

Password-indicated data sets can be selected for migration by using the PASSWORD parameter on the MODESET command. Since conventional tape support does not allow both indicated and non-indicated data sets to be placed on the same tape, this parameter causes ONLY indicated data sets to be scanned for possible selection. If selected and migrated, the original password will be needed to access the tape data set.

If it is desired to process non-indicated data sets as well, a separate job execution is made but without the `PASSWORD` parameter. This technique results in two different series of tapes being created, one with only indicated data sets and the other with only non-indicated data sets.

Reports

The report generated by migrating one or more data sets to tape can be produced in three different sequences:

1. in data set name sequence
2. in data set name sequence within volume
3. in exact processing sequence (unsorted)

By default, the first and second sorted reports are both produced. The third order is produced only if the default sequencing options are changed via sysparm MIGRSORT. Any combination of the three report sequences may be specified.

Restrictions

Sequential data sets with the following attributes are not currently supported:

- USER LABELS — not to be confused with standard labels used on both disk and tape.
- TRACK OVERFLOW — blocks larger than track size.

Condition Codes

The following table is a list of condition codes that may occur while executing Sequential Migrate:

Table 14-2. Sequential Migrate Condition Codes

Code	Description
0	normal completion, no exceptions (execute remaining job steps)
4	exceptions, but some data sets were successfully processed (execute remaining job steps)
8	exceptions, no data sets were processed (bypass remaining job step)

Sequential Migrate Commands

The moving and recataloging of sequential DASD data sets to tape is accomplished by use of the following JCL:

```
//SEQMIGR EXEC MIGRATE  
//SYSIN DD *
```

MODESET, MIGRATE, and SCAN commands as described below. (Only a single MODESET command is permitted and must be first in the command stream. Multiple MIGRATE and SCAN commands may be entered.)

MODESET Command and Parameters

This command is used to restrict the function of all subsequent commands executed in the job step. Only one such command is permitted and, if present, it must be the first in the input stream.

```
MODESET SIMULATE,NORECATLG,NOSCRATCH,PASSWORD,DMYCATLG
```

SIMULATE

The presence of this parameter results in messages being displayed as if migration were taking place. However, no data sets will be written to tape and no catalog updating or scratching will occur.

NORECATLG

Specification of this parameter will deactivate the recataloging of the migrated data sets that would normally be done after their movement to tape.

NOSCRATCH

Specification of this parameter will preclude the scratching of the data sets from DASD after they are migrated.

PASSWORD

Specification of this parameter will cause processing to include only password-indicated data sets. All non-indicated data sets will be bypassed, so the output volume will contain the indicated data sets. Each data set on the output volume will be protected by the same password it had on disk.

DMYCATLG

Specification of this parameter will cause SAMS:Disk to catalog each SAMS:Disk-generated first file. The default is to not catalog these data sets.

MIGRATE Command and Parameters

A single specific data set is explicitly moved from disk to tape by use of the MIGRATE command. More than one of these commands may be included in the input

stream. Explicit requests are not limited by the sysparm options addressed on page 372.

```
MIGRATE DSNNAME= , VOLUME=
```

DSNAME=

The name of the data set to be moved to tape. A relative GDG name is permitted.

VOLUME=

If the system catalog does not contain the serial number of the DASD volume on which the data set resides, this parameter is required.

SCAN Command and Parameters

One or more volumes may be scanned to accomplish the implicit migration of data sets that meet user-specified criteria. The SCAN command and its parameters are described below:

```
SCAN VOLUMES= ,                <-- scan environment parameters
DSNAMES= ,
EXEMPT, METABLE= ,
GDGSONLY, NONGDGS ,
MINSIZE=, MAXSIZE=, TYPE=,      <-- selection criteria test
LASTUSE=, DATE=, GENER=, ANY
```

Note: The volumes= parameter and at least one of the selection criteria (MIN=, MAX=, LAST= or GENER=) is required. If selection based solely on the DSN= list is desired, code MIN=0. MIN=, MAX= and LAST= apply to both GDG and non-GDG data sets. GENER= applies only to GDG data sets. For example, coding GENER= and MIN= on the same command implies two tests for GDGs, one for non-GDGs.

VOLUMES=

This required parameter contains a list of one or more volume serial numbers or prefixes to be scanned. (A MAXIMUM of 50 entries is supported.)

DSNAMES=

A list of one or more data set names or patterns to which migration is to be confined. (A maximum of 10 entries is supported.)

EXEMPT

This parameter causes an exemption list to be examined before migrating any data set. Unless the MET= parameter specifies an alternate member name, the exemption list is assumed to be in parmlib member MIGRATXM.

METABLE=

If an exemption list has been placed in a member other than MIGRATXM, specify the member name as the value of this parameter.

GDGSONLY

The presence of this parameter limits the data sets being considered for processing to only those that have GDG names.

NONGDGS

The presence of this parameter limits the data sets being considered for processing to only those whose names are NOT of the GDG format.

MINSIZE=

Specify the minimum size of a data set that is allowed to be migrated. Data sets equal to or larger than the size given will be selected. This optional selection parameter is specified in units indicated by the TYPE parameter.

- 0 to 3333 is permitted for TYPE=C (cylinders)
- 0 to 99999 is permitted for TYPE=T (tracks)
- 0 to 99999 is permitted for TYPE=K (kilobytes)

MAXSIZE=

Specify the maximum size of a data set that is allowed to be migrated. Data sets less than or equal to the size given will be selected. This optional selection parameter is specified in units indicated by the TYPE parameter.

- 0 to 3333 is permitted for TYPE=C (cylinders)
- 0 to 99999 is permitted for TYPE=T (tracks)
- 0 to 99999 is permitted for TYPE=K (kilobytes)

TYPE=

Specify the type of units in which the size criterion has been expressed in the MIN-SIZE or MAXSIZE parameter.

K—represents Kilobytes (the default type)

C—represents Cylinders

T—represents Tracks

LASTUSE=

This optional selection criterion causes unused data sets to be migrated. Specify the number of days that a data set is permitted to remain on disk without being used. When this number of days has elapsed, the data set is selected for migration.

DATE=

The current date is normally used to determine the number of days since a data set was last used. An alternative date to be used in this calculation may be specified for this parameter. It must be in an accepted SAMS:Disk format.

GENER=

This optional selection criterion is used to specify the number of generations of each GDG data set that are to be kept on disk. All older generations will be selected for migration. For example, GENER=2 is interpreted to mean “keep the two most current generations of each GDG on disk and migrate the others to tape”. In other words, DSN=THE.GDG(0) and DSN=THE.GDG(-1) are kept on disk; the (-2) , (-3) , etc. levels are selected for migration.

Exceptions to the number specified on the GENER= parameter may be supplied in member GDGDASDG. As each GDG data set is being processed, entries in the GDGDASDG member will be searched for a matching data set name or pattern. If found, the number of generations kept on DASD will be the number given in the member entry. If no match is found, the global value supplied by the GENER= parameter will be used. Refer to the description under the topic *"GDG DASD Generations List"* on page 450 in the *Systems Guide* for more details.

Note: The GDGDASDG member is not searched unless the GENER= parameter is present.

ANY

When more than one selection test is specified, this parameter causes the data set to be migrated if ANY single test is satisfied. The default requires that ALL multiple tests be satisfied.

Migration Concerns

A duplicate copy tape for backup purposes may be created concurrently with the primary tape. The following installation requirements must be met for this capability to be used:

- sysparm MIGPSCOP must be specified with a value of Y.
- the //MIGRATEC dd statement must be in the JCL.
- the files data set must have the MIGRECAT subfile properly initialized

Consult your local SAMS:Disk installation and support personnel to determine if this option is being used, or to request a change.

The MIGRECAT file is used to contain a record for each data set that is migrated to a primary and copy tape. When a primary tape volume cannot be used, the RECATALOG command is used to read these records and recatalog all of the data sets on the primary volume to the backup volume.

It must also be noted that SAMS:Disk does not provide any formal automated means for managing the entries in the MIGRECAT file. SAMS:Disk has no way of knowing how long to keep them. For this reason, the file will need to be purged occasionally to keep it from becoming excessively large. To accomplish a purge of this (or any subfile), merely unload the files data set, excluding MIGRECAT (filename) from the list of files, and then reload the unloaded records. Turn to page 17 in the *Systems Guide* for more information on *UNLOAD* and *RELOAD*.

Catalog Concerns

Also be aware that catalogs will fill up unless some measure is taken to purge unwanted entries. This can be aided by limiting the migration function to GDG data sets. As new generations are created, the older generations cataloged to tape will automatically be uncataloged as part of normal GDG processing.

Recatalog JCL - Recatalog to Duplicate Tape

The following JCL is required to recatalog data sets from the primary to the backup volume:

```

\\RECAT   EXEC  PGM=ADSMI002,PARM=ADSDM272,REGION=1024K
\\STEPLIB DD   DISP=SHR,DSN=SAMS.DISK.LOADLIB
\\ABNLDUMP DD   DUMMY
\\CMDPRINT DD   SYSOUT=A
\\FILES    DD   DISP=SHR,DSN=SAMS.DISK.FILES
\\MSGPRINT DD   SYSOUT=A
\\PARMLIB  DD   DISP=SHR,DSN=SAMS.DISK.PARMLIB
\\SYSPRINT DD   SYSOUT=A
\\SYSUDUMP DD   SYSOUT=A
\\SYSIN    DD   *

```

Figure 14-1. RECATALOG JCL

RECATALOG Command—Parameter Descriptions

```
RECATALOG OLDVOLNO=,SIMULATE
```

OLDVOLNO=

The volume serial of the original volume that is no longer available.

SIMULATE

The presence of this keyword causes messages to be produced as if recataloging was taking place when, in fact, no catalog changes occur.

Sample MIGRATION Report

1987.338	DEC 04, 1987	S E Q U E N T I A L D A T A S E T M I G R A T I O N													PAGE	1
FRIDAY	8.14 AM	DATA SET NAME/VOLUME SEQUENCE													SAMS:Disk	8.2
		DEVICE		REC		TAPE										
DATA SET NAME		VOLUME	TYPE	EXPDATE	FM	BLKSZ	LRECL	OPTCD	KEYL	VOLUME	SEQ	SCR	CAT	REAS	KBYTE	TRACKS
-----		-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----
LABS.KSV.ASMLOG		LABS83	3380		F	132	132	00	0		4	Y	REC		142	3
LABS.KSV.LSYSPRIN		LABS83	3380		FB	4,840	121	00	0		3	Y	REC		47	1
LABS.KSV.MTFTABLE		LABS83	3380		VB	2,994	2990	00	0		2	Y	REC		94	2
LABS.KSV.SYSTERM		LABS83	3380		FB	4,840	121	00	0		1	Y	REC		94	2
-----		-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----
TOTAL DATA SETS MIGRATED		4												TOTAL KILOBYTES		377
												TOTAL TRACKS				
REASON CODES: M=MINSIZE X=MAXSIZE L=LASTUSE G=GENER																

1987.338	DEC 04, 1987	S E Q U E N T I A L D A T A S E T M I G R A T I O N													PAGE	1
FRIDAY	8.14 AM	VOLUME/DATA SET NAME SEQUENCE													SAMS:Disk	8.2
		DEVICE		REC		TAPE										
VOLUME DATA SET NAME		TYPE	EXPDATE	FM	BLKSZ	LRECL	OPTCD	KEYL	VOLUME	SEQ	SCR	CAT	REAS	KBYTE	TRACKS	
-----		-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	
***** THIS IS A SIMULATION RUN																
LABS83 LABS.KSV.ASMLOG		3380		F	132	132	00	0		4	Y	REC		142	3	
LABS83 LABS.KSV.LSYSPRIN		3380		FB	4,840	121	00	0		3	Y	REC		47	1	
LABS83 LABS.KSV.MTFTABLE		3380		VB	2,994	2990	00	0		2	Y	REC		94	2	
LABS83 LABS.KSV.SYSTERM		3380		FB	4,840	121	00	0		1	Y	REC		94	2	
-----		-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	
TOTAL DATA SETS MIGRATED		4												TOTAL KILOBYTES		377
												TOTAL TRACKS		8		
REASON CODES: M=MINSIZE X=MAXSIZE L=LASTUSE G=GENER																

Figure 14-2. Sample MIGRATION Report

Chapter 15. DASD Billing

This SAMS:Disk facility provides billing capabilities for the space occupied by non-VSAM data sets and VSAM clusters on DASD. Periodic VTOC scans are made to determine the space occupied by each individual data set. Use assigned billing rates are then applied and accumulated. The VTOC scans may be run as frequently as desired. Each run's billing amount is based on current space occupancy assumed to have existed over the period since the previous scan.

Commands Available:

ACCUMULATE
BILLING
EXCLUDE
EXTEND
PASSWORD
SCAN
SELECT

JCL Proc Executed:

BILLING, VSAMBILL
DMS
DMS
EXTEND
DMS
DMS
BILLING
SELECT
DMS

Special Parmlib Members:

DSBCATEG
DASDSPCB
USERCODT
DSBALTRT

Special Files:

DSBUSERD
DSBEXCLU
DSBRATEC

General Information

The SAMS:Disk DASD Space Billing function accumulates a billing charge against each data set based on two criteria:

1. The SPACE OCCUPIED by (allocated to) each data set
2. The NUMBER OF DAYS it occupies that space.

The product of these two values provides a billing quantity in either TRACK-DAYS or KILOBYTE-DAYS. The user must supply the billing rate for either one track-day or one kilobyte-day of occupancy. A simple multiplication of the billing quantity times the billing rate yields the billing charge for each data set. For example, a two-track data set residing on DASD for 50 days provides a 100 track-day billing quantity. If the billing rate is one cent per track per day, the charge for this period is one dollar.

To find the space occupied by each data set (item 1 above), periodic VTOC scans are made and space allocations calculated.

There are two methods available to determine the days of occupancy for data sets. They are referred to as MODE=ELAPSED and MODE=IMMEDIATE.

Elapsed Mode

With MODE=ELAPSED a data set is not charged for a day of occupancy until it exists on the volume one full day. The period of occupancy (item 2 above) is computed by counting the number of days from the current run date back to one of three dates described below, whichever is most recent.

1. The date of the last previous billing scan
2. The creation date of the data set,
3. A BEGIN= date supplied on the billing command.

SAMS:Disk assumes that the current space occupancy has existed over the period computed. The advantage of ELAPSED mode processing is that a user gets billed for DASD space only after the data set exists for a full day. Disadvantages include that transient data sets do not ever get billed, and you must run two billing steps on the first day of each cycle, one to complete the last cycle and one to start the next. When using this mode, watch for data sets that are deleted and reallocated each day. Depending on the time of the billing run, they may never have charges accumulated for them.

Immediate Mode

IMMEDIATE mode indicates that a data set is charged for a day of occupancy each time it is encountered during a Billing Accumulate run. The maximum is once per day, but the data set need not exist for a full day before it starts accumulating charges.

The advantage of IMMEDIATE mode of processing is that any data set existing at the time of the billing run has charges accumulated for it, and there is no need to make a special run on the first day of a billing cycle to close out the prior cycle. The biggest disadvantage is that a data set may have existed only during the time the billing run is executed and will get a full day's charge against it. This technique also catches data sets scratched and recreated each day.

Billing Rates

The user must select a billing rate for each allocated track or kilobyte per day. Different billing rates can be selected and assigned to different volumes or groups of volumes. As an example, TSS volumes may be billed differently from WORK volumes, or 8595 volumes differently from 8598 volumes. All volumes to be billed at the same rate must be assigned a CATEGORY CODE in member DSBCATEG of the parmlib data set. The associated billing rate for each category code must then be entered into parmlib member DSBRATEC.

Basically, a category code is a code assigned to a group of volumes that are to be assigned the same per-day billing rate.

Computing the Billing Charge

As each volume is scanned, its matching category code is retrieved from member DSBCATEG. This code is then used to find the proper rate in member DSBRATEC. Each data set's allocated space and billing period are computed, the rate is applied, and the result is added to the accumulating charge. These charges are stored and updated for each individual data set. The records are placed in the DASDSPCB subfile of the files data set. The VTOC SCAN and ACCUMULATE runs may be executed as often as needed to achieve the data set sampling required.

Assign Volumes to Billing Categories

Volumes that are to be assigned the same billing rate must be given a category code and placed in member DSBCATEG in the parmlib data set. The format of entries in this member is:

CCVVVVVV

Where: CC is the CATEGORY CODE,
VVVVVV is a VOLSER or PREFIX

Define Billing Rates for Each Category

Billing rates for each category code must be placed in member DSBRATEC of the parmlib data set. The billing rate is the amount to be charged to the user for each 1000 bytes of DASD space allocated to a data set each day, or the amount to be charged to a user for each track of space used each day. This depends on sysparm DSBUNITS.

The format of entries in this member is:

CCRRRRRR

Where: CC is the CATEGORY CODE
RRRRRR is the ASSOCIATED BILLING RATE.

The category code in the DSBCATEG member is matched to those in the DSBRATEC member to find a rate to apply to the volume being processed.

The rate specified by RRRRRR may be a 1- to 5-digit value if the decimal point is also specified, or a 6-digit value if the decimal point is omitted. Whenever you omit the decimal point, SAMS:Disk right-justifies the rate; that is, zeros are added on the left until all six positions are filled. SAMS:Disk then assumes that the decimal point is to precede the first digit (for example, a 4-digit value of 0621 means .000621, not .0621!). You can easily avoid this minor degree of confusion, however, by always specifying the decimal point in the rate field, or always supplying a full character field. The decimal point may be placed anywhere in the RRRRRR field.

The billing rate defaults to being interpreted as a dollars and cents value. To use currencies other than the dollar, see *“Using Other Currencies”* on page 389.

Example of Assigning Billing Rates

A billing rate of \$ 0.008500 is to be applied to data sets on volumes VOL001, VOL006, and VOL007. A billing rate of \$ 0.007800 is to be applied to volumes VOL002 and VOL004. All volumes beginning with TSS are to be billed at a rate of \$ 0.005562.

Category codes are defined in member DSBCATEG:

- '12VOL001'
- '12VOL006'
- '12VOL007'
- '13VOL002'
- '13VOL004'
- '20TSS/XX'

Billing rates for each category are defined in member DSBRATEC:

- '12008500'
- '13.0078'
- '20005562'

Billing Units System Parameters

The billing unit, either track-days or kilobyte-days, is indicated by the following system parameter.

DSBUNITSuuu - system parameter

Specify TRK for the appended uuu if billing is to be done by track-days rather than by the default of kilobyte-days. K/B may be used to specify the default, but it isn't needed.

Using Other Currencies

If you desire to accumulate and charge for DASD space in a unit of currency other than dollars and cents, such as Japanese yen or Italian lire, specify sysparm DSBALTRT with a value of 1. This changes the billing rates assigned in the DSCRATEC member from a dollars and cents value to just a unit of currency expressed with decimal places. If this feature is used, specifying 993.25 in the DSBRATEC member would indicate that for all volumes assigned to category code 99, either 3.25 lire or 3.25 yen are to be charged for each track or kilobyte day. (To review the rules explaining the format of the DSBRATEC table entries, see “*Define Billing Rates for each Category*” on page 387.) The rate value shown on the billing reports will be shown as five digits with two decimal places, but the amounts and totals will be shown as full numbers with no decimal places.

Note: that this alternate rate specification may also be used for other forms of currency as appropriate.

User Code/Account Codes

When a scan and accumulate run encounters a data set for the first time (no prior billing entry), a user/account code is assigned that becomes a part of its billing record in the DASDSPCB subfile. The user code assigned is significant in that billing reports are grouped and subtotaled for each unique user code. The default user code is the first 12 bytes of the data set name, but it may alternately be specified in any of the following ways:

1. With a displacement into the format-1 DSCB where a user code can be found.
2. By providing a list of user codes matched to data set names in member USERCODT of the parmlib data set.
3. As one of the index levels of the data set name.

Since the user code is assigned and placed in the billing subfile record the first time the data set is processed, the manner used to assign user codes should be implemented prior to running DASD billing; that is, user code assignments cannot be dynamically updated by changing the method by which they are assigned. To change an assigned user code, either user exit DSBUSREX must be used, or the billing subfile must be reformatted and DASD billing rerun. (Reformatting destroys all existing records, which necessitates the rerun.)

A 40-character description may optionally be associated with each user code. It will be printed next to the user code when the billing totals are produced.

Assigning User Codes

The first item that controls the assignment of a user code is sysparm DSCBUSRD. It is used to provide the displacement into the DSCB to find the user code field. If this displacement is zero, which is the default, an attempt is made to find a user code in member USERCODT. If this member is defined, it may supply the code. But by default it is not defined and processing returns. When this member check returns and no code has been assigned, either the DSCBUSRD sysparm-defined displacement is used to find the user code, or an index level of the data set name is used. Since the index level technique must be activated specifically by sysparm DBDSNINX, the resulting default is to use the displacement method. Each of the possible methods is defined in more detail below.

Method 1: Displacement

DSCBUSRDxxx - system parameter

The displacement of the user code in the DSCB must be specified as the xxx value. The default value is 000.

USRCLENxx - system parameter

The length of the user code in the DSCB must be specified as the xx value. The default (and maximum allowed) is 12.

When the value of the displacement is zero, an attempt is made to find a user code in member USERCOTD of the parmlib data set. If the displacement is non-zero, the member is never inspected.

Method 2: PARMLIB Member USERCOTD

An attempt to find the user code in this member is made only if the displacement is zero. Entries in this member have the following format:

```
'CCCCCCCCCCCCDDDD...DDDD'
```

Where: CCCCCCCCCCCC is the 12-character user code
 DDDD....DDDD is a full data set name or a data set name pattern

If the CCC field is less than 12 characters, pad it with blanks out to the DDD field and enclose the entire entry in quotes.

Method 3: Index Level (Node)

DBDSNINXn - system parameter

Change the appended N to Y to indicate that the user code is an index level from the data set name. The default value is N, indicating that the user code is found by sysparms DSCBUSRD and USRCLEN.

DBDSNCFDx - system parameter

Specify for the appended x the index level of the data set name that is to be used as a user code. The default value is 1, but activated only if sysparm DBDSNINX is specified with a value of Y.

User Code Description Table

Descriptions may be defined for each user code by placing them in member DSBUSERD of the parmlib data set as follows:

```
DSBUSERD MEMBER
```

The format of entries in this member is:

```
'CCCCCCCCCCCCXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
```

Where: CCCCCCCCCCCC is the 12-character user code
 XXX.... is a 40-character description for the user code

If the user code is less than 12 characters in length, pad with blanks and enclose the entire entry in quotes.

Total Charges

Charges accumulated over a billing period, perhaps daily runs over a month's time, can be totaled (extended) as desired. This causes the accumulated charges to be reset to zero in order to begin the accumulation for a new billing period. Detailed charges are printed for each data set as well as subtotals by user codes and totals by volume and device type.

Exempting Data Sets

Member DSBEXCLU in the parmlib data set may be used to exempt data sets from the billing process. Simply enter their full data set names or matching patterns into the member. Model DSCBs are always excluded by default.

Note: The DSBEXCLU parmlib member only excludes data sets from the data accumulation phase of the Billing process, and not during the extend reporting job. It is therefore possible to exclude data sets from billing for a portion of the billing cycle, and still have them show up on the report for those days that billing was active for the data sets.

Example entries in SAMS.DISK.PARMLIB(DSBEXCLU):

```
SAMS.DISK.LOADLIB
SAMS.DISK.FILES
SAMS.DISK.PARMLIB
SYS /
```

Deferred Write Precautions

Due to the large number of records processed, the interface to the DASDSPCB subfile of the files data set uses deferred-write support. This technique buffers in memory all updates to a logical block before writing it back out again. This reduces I/O and improves performance considerably, but also eliminates the file's integrity against concurrent updates by other SAMS:Disk tasks. To protect against possible file overlays, ensure that DASD Billing is not run when any other SAMS:Disk task that updates the files data set is also running. The following suggestions are given to provide the needed protection.

1. In a SINGLE-CPU environment, DISP=OLD for the files data set in the BILLING JCL PROCEDURES will suffice.
2. In the MULTIPLE-CPU/SHARED-DASD environment, and as an alternative even in the single-CPU environment, isolate DASD billing from all other SAMS:Disk functions by giving it its own unique files data set; for example, DSN=SAMS.DISK.BILLING.FILES. Modify the distributed JCL procedures BILLING and EXTEND to reference this files data set. Also establish unique JCL to format, unload, and reload this files data set. The same parmlib data set can be used as for other SAMS:Disk functions, but you may want to consider a separate one just for the format, unload, and reload JCL such that the FILEDEFN member can be changed to eliminate the definitions of all the other unneeded subfiles. Then the deferred write feature is of no concern and DASD billing can be run independently from the other SAMS:Disk functions.
3. A third technique is to simply schedule DASD billing when no other SAMS:Disk functions are executing.

DASD Billing Subfile Capacity

The FILEDEFN entry for the DASDSPCB subfile should be examined for proper capacity and modified if necessary. To activate the changed definition, an existing files with entries to be saved must be reorganized; that is, unloaded and reloaded. If no entries need to be saved, the files can just be reformatted. If a unique files data set for billing is NOT being used, care must be taken to unload and reload the entries from all subfiles, not just the DASDSPCB subfile.

User Exits

Specifics regarding the use of user exits can be found beginning on page 195 of the *Systems Guide*. Only a brief description is presented here.

DSBUSREXmmmmmmmm

This exit gives control to the user program when every billing record is being inserted or updated. One use of this exit is to reset a previously assigned user code. For complete details of this user exit, please turn to page 135 in the *Systems Guide*.

USRCODEXmmmmmmmm

This exit gives control to the user prior to looking in member USERCODT for the user code to be assigned. The exit may supply the user code directly. For complete details of this user exit, please turn to page 180 in the *Systems Guide*.

Condition Codes

The following table is a list of possible condition codes that may occur during the execution of any PROC related to DASD Billing:

Table 15-1. DASD Billing Condition Codes

Proc Step	Code	Description
Select	0	Successful completion
	4	Command rejected
Accumulate	0	Successful completion
	4	Unable to update one or more records in billing file
	8	Command rejected
	16	Input data set from sort could not be opened
Extend	0	Successful completion
	4	Nothing could be totaled (no billing data)
	8	Command rejected

Accumulate JCL for Non-VSAM Data Sets

The following JCL is required to accomplish the scan and accumulation for billing records placed in the files data set:

```
//DAILY EXEC BILLING
//SELECT.SYSIN DD *           <- NOTE use of STEP NAME
//ACCUM.SYSIN DD *           <- NOTE use of STEP NAME
```

The SELECT and ACCUMULATE Commands are described on the following pages.

SELECT Command and Parameters

The following command is used to scan DASD volumes and retrieve the DSCB for each data set on the volume:

```
SELECT VOLUME=,DSNAME=
```

VOLUME=

A list of one or more volumes or volume prefixes to which scanning is to be confined. (A maximum of 100 entries is supported.)

DSNAME=

A list of one or more data set names and patterns to which the processing is to be limited. (A maximum of 25 entries is supported.)

ACCUMULATE Command and Parameters

The following command is used to accumulate the billing charges for the data sets on each DASD volume:

```
ACCUMULATE    DATE=,MODE=,BYPASSNEW,ADDMISSING,BEGIN=,
              BWEEK=,BMONTH,BMEMBER=,SIMULATE
```

DATE=

A date other than the current date may be substituted for the current date. This allows users to run billing as if it had been run on a different date. The date may be in any accepted SAMS:Disk format.

Note: Data sets with a creation date greater than this date will be bypassed.

MODE=

Two different techniques of billing are supported. They are selected through the MODE= parameter. When MODE=IMMEDIATE is specified, occupancy days are calculated each day a data set is found during the ACCUMULATE process; that is, if a data set is seen today, it will be billed for the day.

When **MODE=ELAPSED** is specified, occupancy days are calculated based on the number of days the data set has existed since the last **ACCUMULATE** process.

The primary difference between these two modes is that **IMMEDIATE** calculates occupancy days immediately, as soon as a data set is found on a volume, while **ELAPSED** requires that the data set exist across **ACCUMULATE** runs on two different days before charges will be accumulated.

BYPASSNEW

Valid only if **MODE=IMMEDIATE** is specified, the **BYPASSNEW** parameter indicates that if a data set's creation date is the same as the current date, no occupancy days are to be accumulated for it. The purpose of this command is to provide a way to avoid billing for transient data sets (those that are on the volume less than one day).

ADDMISSING

Valid only when **MODE=IMMEDIATE** is specified, the **ADDMISSING** parameter indicates that processing is to calculate occupancy days for all of the days the data set existed since the **Accumulate** processing was previously run.

If **Accumulate** processing has not previously processed a data set in this billing cycle and the **ADDMISSING** parameter is specified, occupancy days will be determined by the date of the beginning of the billing cycle or the create date of the data set, whichever is later. If not specified, only one occupancy day is calculated each day the **accumulate** is run.

BEGIN=

This date is used to determine the beginning date of the billing cycle. No data sets will be billed for occupancy days before this date. The date may be in any accepted **SAMS:Disk** format. (See **BWEEK=**, **BMONTH**, and **BMEMBER=** parameters also)

BWEEK=weekday

For users implementing a weekly billing cycle, the **BWEEK=** parameter is an alternative to the **BEGIN=** parameter for setting the begin date of the billing cycle. Specify the name of a day in the week, and the beginning of the billing cycle will automatically be set each week to the date corresponding to the specified weekday name; for example, **BWEEK=MONDAY**.

BMONTH

For users implementing a monthly billing cycle, the **BMONTH** parameter is an alternative to the **BEGIN=** parameter for setting the begin date of the billing cycle. When **BMONTH** is specified, the beginning of the billing cycle will automatically be set each month to the date corresponding to the first day of the month.

BMEMBER=

This parameter is an alternative to the BEGIN= parameter for setting the begin date of the billing cycle. The BMEMBER= specifies a member name in the parmlib that contains begin dates of billing cycles. When BMEMBER= is specified SAMS:Disk scans the specified member name to determine the begin date of the active billing cycle.

The entries must be in an accepted SAMS:Disk format. Sample entries in the parmlib member:

- 01/01/95
- 04/01/95
- 07/01/95
- 10/01/95
- 01/01/96

To determine the BEGIN date to use, SAMS:Disk scans the member entries searching for the highest date that is equal to or prior to the current date. For example, using the above entries, if the date of the accumulate run is 02/15/95, the Begin date used would be 01/01/95.

To use this parameter, first create a new member in the parmlib data set with the dates corresponding to your billing cycle begin dates. Then specify the member name in the BMEMBER= parameter.

SIMULATE

The presence of this parameter will cause the processing to take place, but no billing records will actually be updated in the files data set.

ACCUMULATE JCL for VSAM Clusters

The following section describes the JCL and control statements needed to run the BILLING functions against VSAM clusters. It is equivalent to the SCAN and ACCUMULATION process used for non-VSAM data sets. This is the only portion of the billing function that is unique to VSAM cluster billing.

Keep in mind that the EXTEND procedure will process both the VSAM and non-VSAM records.

```
//jobname JOB (acct,info),etc.
//DAILY EXEC VSAMBILL
//DMS.SYSIN DD *          <-- NOTE use of STEP NAME
* Specify the following DSCL commands for VSAM billing.          *
    PASSWORD MASTER=,CATALOG=                                (optional)
    SCAN CATALOGS=/                                           (required)
    EXCLUDE CRITERIA=(DSORG,NE,VSAM)                          (required)
    SELECT CRITERIA=,DSN=,VOL=,TABLE=                         (optional)
    EXCLUDE CRITERIA=,DSN=,VOL=,TABLE=                         (optional)
    BILLING                                                    (required)
//ACCUM.SYSIN DD *          <-- NOTE use of STEP NAME
* ACCUMULATE COMMAND as described for non-VSAM                *
```

Figure 15-1. Sample BILLING Command JCL

A sample job to accumulate VSAM billing records is as follows:

```
//jobname JOB (acct,info),etc.
//DAILY EXEC VSAMBILL
//DMS.SYSIN DD *
    SCAN CATALOGS=/
    EXCLUDE CRITERIA=(DSORG,NE,VSAM)
    BILLING
//ACCUM.SYSIN DD *
    ACCUMULATE BMONTH,MODE=IMMEDIATE,BYPASSNEW
```

Figure 15-2. Sample ACCUMULATE Command JCL

DASD Billing Total JCL

The following JCL is required to generate (extend) the billing totals at the end of a billing cycle:

```
//TOTALS EXEC EXTEND
//SYSIN DD *
EXTEND COMMAND and PARAMETERS as described below
```

Note: In order to run the EXTEND JCL, a restart data set must be allocated. Please refer to the topic "*Activating DASD Billing*" on page 27 in the *Installation Guide* if this has not been done.

This command causes billing records and reports to be produced and is used at the end of a billing cycle:

```
EXTEND VOLUME=,DSNAME=,SIMULATE
```

VOLUME=

A list of one or more volumes and volume prefixes to which processing is to be confined. Omission of this parameter results in no exclusion on the basis of volume. (A maximum of 50 entries is supported.)

DSNAME=

A list of one or more data set names and patterns to which the run is to be limited. (A maximum of 50 entries is supported.)

SIMULATE

Inclusion of this parameter will cause the reports to be produced without the updating of the billing records.

Implementation - Suggested Run Procedures

The following sequence of steps is suggested as an approach to doing regular DASD BILLING on a monthly billing cycle:

1. Allocate and initialize a separate files data set just for DASD billing.
2. Specify the BMONTH parameter in the ACCUMULATE command to implement a monthly billing cycle that sets the begin date automatically to the first day of each month. Also, specify MODE=IMMEDIATE to calculate occupancy days for each day a data set is found during the ACCUMULATE processing.
3. Execute the SCAN and ACCUMULATE job once a day, every day, during the monthly billing cycle.

For example, August has 31 days so SCAN and ACCUMULATE should be run for 31 days.

4. At the end of the month (the last day):
 - a. Unload the billing subfile (FILE=DASDSPCB) to create a sequential backup copy.
 - b. Run the EXTEND job to generate the billing totals, print the reports, and zero the accumulation values. (This may be in simulate mode if step (c) is done.)
 - c. Reformat the billing subfile to destroy all the billing records. If a separate files data set is used for billing, it can simply be reformatted. If the files data set is used jointly with other SAMS:Disk functions, the DASDSPCB subfile should be EXCLUDED during a files reorganization (unload and reload). If this is not done, all data sets that existed one month but not the next will continue to appear, but with zero billing amounts. (For more information on unloading, formatting and reloading the files data set, turn to page 17 in the *Systems Guide*.)

Billing Processing Considerations

If a data set is scratched during a billing period and is later recreated or restored, the creation date of the new data set determines how charges are accumulated for it if MODE=ELAPSED, or ADDMISSING is being used.

When a data set is restored with the SAMS:Disk Restore function or moved with the Move/Copy function, it is by default given its original create date. For the purpose of determining the data sets period of occupancy on DASD, this old date is misleading and can result in overcharges.

There are two options to help avoid this overcharging:

- Option 1 — Use the `CREATE` parameter in either `Restore` or `Move/Copy` functions. This parameter sets the creation date to the current date.
- Option 2 — Specify the `MODE=IMMEDIATE` without the `ADDMISSING` parameter in the command, and the data set will be charged for only one day no matter when it was restored. If the creation date is greater than the creation date of the original data set, charges will be accumulated for only the days the data set actually existed.

Note: If the creation date of the restored data set is the same as the creation date of the original data set before it was scratched and the `ADDMISSING` parameter is specified, the data set will be charged for all days in between the two creation dates, even if it did not exist on the volume during that time.

DASD Billing and Archival Functions

If both DASD Billing and archiving functions are being used, it is probably desirable to run the billing accumulations just prior to `ARCHIVING`. If this is not done, the “offending” data sets escape their last billing.

If an abnormal termination occurs at any point during the `SCAN` or `ACCUMULATE` functions (`BILLING JCL` procedure), you may restart processing by simply rerunning the entire procedure. This is possible because `SAMS:Disk` will not accumulate charges for a data set that has already been processed on the same day. Some data sets will be reanalyzed, but no additional charges will be accumulated for them.

If an abnormal termination occurs at any time during the extend function (`EXTEND JCL` procedure), you may simply rerun it. This is possible because of the restart data set associated with extend processing. (If a separate files data set is being used for billing and a backup was taken prior to the extend run, you may also restore the files to its prior status, delete and reallocate the restart data set, and start the entire process again, if desired.)

Increased Billing Buffers

A default buffer size to hold entries in memory is controlled by the following sysparm. If this default value is not large enough due to the number of entries placed in the member, a diagnostic message is issued. The sysparm must be changed to increase the buffer size.

DSBCATSZxxxxx

The buffer size for the category codes. The default value is 00500.

Definitions of Fields in the DASD Billing Report

USER CODE

The user code is the account code used in DASD Billing to group particular data sets for a billing report. It is specified by the user in the USERCODT member of parmlib or by use of format-1 DSCB displacement or index level processing.

DESCRIPTION

This is descriptive information for the user code. It is entered in the DSBUSERD member of parmlib, and is used for printing purposes only.

DATA SET NAME

The name of the data set for which billing is being done.

VOLUME

The volume on which the data set occupies space.

DEVICE TYPE

The type of device of the volume.

CREDIT

The creation date of the data set (Julian format).

USEDT

The date the data set was last used (Julian format).

LAST SCAN

The date that billing ACCUMULATE processing was last run for the data set.

ALLOC

The number of tracks that are allocated to the data set.

Note: Please note that this allocation figure represents the size of a data set at the time it was last examined. Unless it is a very static data set, its size may change from day to day. This circumstance will make it difficult for you to check the K/B DAYS or TRK DAYS field for accuracy through a manual calculation unless you already know that the data set has remained static in its allocation. The K/B DAYS or TRK DAYS fields, although correct in the report, may be misleading in such cases. Please keep this in mind when looking at your report.

MAX USED

The number of tracks that are actually being used by a data set as opposed to the number of tracks that are allocated.

K/B DAYS

The total number of kilobytes allocated to the data set multiplied by the number of days' occupancy. The algorithm is as follows:

$$(\text{TRACKS} * \text{TRACK SIZE}) * \text{NUMBER OF DAYS OCCUPANCY} / 1000$$

TRK DAYS

The total number of tracks allocated to the data set multiplied by the number of days' occupancy. The algorithm is as follows:

$$\text{TRACKS} * \text{NUMBER OF DAYS OCCUPANCY}$$

RATE

The rate of charge (per day) to be applied to a data set. The format of this field depends on which value is specified for sysparm DSBALTRT. If it is specified with a value of N, standard rate processing will be used and the format will be whole numbers. If it is specified with a value of 1, foreign currency rate processing will take effect and the format of the field will have a decimal point. (The last type of process is intended for currencies such as the Japanese yen or Italian lire.)

AMOUNT

The total amount that is charged for the data set during this billing cycle.

TOTAL DATA SETS

The total number of data sets being billed in the billing cycle.

TOTAL ALLOCATED TRACKS

The total track allocation of all data sets that were billed in the billing cycle.

TOTAL K/B DAYS

The total number of K/B DAYS for all data sets in the billing cycle.

TOTAL TRK DAYS

The total number of TRACK DAYS for all the data sets in the billing cycle.

TOTAL AMOUNT

The total amount charged for all data sets in the billing cycle.

SAMS:Disk Arithmetic Restriction

SAMS:Disk DASD Billing uses fullword arithmetic in its calculations for totals. This restricts the highest number to hex 7FFFFFFF or decimal 2,147,483,647. If the number of K/B DAYS accumulated in a cycle is higher than this, the totals will be incorrect. However, the grand totals on the report will be correct.

DASD Billing Samples

The following pages contain sysout Billing samples sorted by:

1. Device Type
2. User Code
3. Volume

Sample by Device Type

1993.039 FEB 08, 1993		DASD BILLING BY DEVICE TYPE			PAGE 1
MONDAY 6.20 PM					SAMS:Disk 8.2
DEVICE TYPE		VOLUMES	DATA SETS	TRK DAYS	TOTAL CHARGE
3350		4	6	7112	213.36
3380		3	15	2380	68.77

TOTAL DEVICES	2	TOTAL VOLUMES	7		
		TOTAL DATA SETS	21		
		TOTAL TRK DAYS		9492	
		TOTAL CHARGES			282.13

Figure 15-3. Sample Billing Report - By Device

Sample by User Code

1993.039 FEB 08, 1993	D A S D B I L L I N G B Y U S E R C O D E							PAGE 1
MONDAY 6.20 PM								SAMS:Disk 8.2
USER CODE - LABS	DESCRIPTION - LABS GROUP							
		DEVICE				-----TRACKS-----		
DATA SET NAME	VOLUME	TYPE	CREDATE	USEDATE	LAST SCAN	ALLOC	MAX USED	TRK DAYS RATE AMOUNT
LABS.MJB.ASMLOG	LABS83	3380	1987.020	1987.336	1987.341	35	35	630 030000 18.90
LABS.MJB.DRIVE	LABS82	3380	1983.054	1987.335	1987.341	25	15	175 030000 5.25
LABS.MJB.FILES	LABS83	3380	1983.102	1987.252	1987.341	30	30	210 030000 6.30
LABS.MJB.ISPMLIB	LABS83	3380	1983.116	1987.341	1987.341	4	3	28 030000 .84
LABS.MJB.ISPPLIB	LABS83	3380	1983.115	1987.341	1987.341	3	2	21 030000 .63
LABS.MJB.ISPSLIB	LABS83	3380	1983.124	1987.341	1987.341	2	1	14 030000 .42
LABS.MJB.ISPTLIB	LABS83	3380	1986.329	1987.341	1987.341	3	2	21 030000 .63
LABS.MJB.JCL	LABS83	3380	1983.003	1987.341	1987.341	30	23	210 030000 6.30
LABS.MJB.LOADLIB	LABS82	3380	1987.133	1987.341	1987.341	24	16	168 030000 5.04
LABS.MJB.LSYSPRIN	LABS83	3380	1983.002	1987.336	1987.341	1	1	7 030000 .21
LABS.MJB.MERGPRIM.DMSC. D0087267.T091603	LABS56	3350	1987.267		1987.341	474	474	3318 030000 99.54
LABS.MJB.MERGPRIM.DMSC. D0087267.T091735	LABS56	3350	1987.267		1987.341	462	462	3234 030000 97.02
LABS.MJB.MERGPRIM.DMSC. D0087267.T092008	LABS53	3350	1987.267		1987.341	40	40	280 030000 8.40
LABS.MJB.MITABLE	LABS83	3380	1987.329		1987.341	3	3	21 030000 .63
LABS.MJB.MVOL.PS	LABS53	3350	1987.258	1987.258	1987.341	8	8	56 030000 1.68
LABS.MJB.MVOL.PS	LABS50	3350	1987.258	1987.258	1987.341	16	16	112 030000 3.36
LABS.MJB.MVOL.PS	LABS51	3350	1987.258	1987.258	1987.341	16	16	112 030000 3.36
LABS.MJB.PARMLIB	LABS83	3380	1983.036	1987.328	1987.341	24	13	168 030000 5.04
LABS.MJB.SEQFILES	WRK800	3380	1987.327		1987.341	15	1	105 005000 .52
LABS.MJB.SOURCE	LABS83	3380	1982.343	1987.338	1987.341	84	71	588 030000 17.64
LABS.MJB.SYSTEM	LABS83	3380	1983.196	1987.336	1987.341	2	1	14 030000 .42
TOTAL DATA SETS 21						TOTAL ALLOCATED TRACKS 1301		
						TOTAL TRK DAYS 9492		
						TOTAL AMOUNT 282.13		

Figure 15-4. Sample Billing Report - By User Code

Sample by Volume

1993.039 FEB 08, 1993	D A S D B I L L I N G B Y V O L U M E							PAGE 1
MONDAY 6.20 PM								SAMS:Disk 8.2
VOLUME	DEVTYPE	NO SCANS	LAST SCAN	DATA SETS	TRK DAYS	TOTAL CHARGE		
LABS50	3350	1	87341	1	112	3.36		
LABS51	3350	1	87341	1	112	3.36		
LABS53	3350	1	87341	2	336	10.08		
LABS56	3350	1	87341	2	6552	196.56		
LABS82	3380	1	87341	2	343	10.29		
LABS83	3380	2	87341	12	1932	57.96		
WRK800	3380	1	87341	1	105	.52		
TOTAL VOLUMES 7			TOTAL DATA SETS 21					
			TOTAL TRK DAYS 9492					
			TOTAL CHARGES 282.13					

Figure 15-5. Sample Billing Report - By Volume

Chapter 16. TSS Command Processor

Command Processors:

DARCHIVE
LISTREQ
DRESTORE
DERASE
RESTORE
LISTDMS

TSS Help Text:

Available for all

Special PARMLIB Members: Special Files:

TSOUSERI

ARCHCMDS
DMSPARMS

RESCMDS
RETEXCLD

General Information

Specialized support has been provided for the TSS environment in the form of command processors. To minimize the amount of familiarization necessary for TSS users to master the commands, the parse service has been used in the command processors for command and parameter formatting and validation.

Use of the TSS command processors requires some special installation tasks to be accomplished. Among the steps are the optional changing of the default names for the SAMS:Disk files and parmlib data sets, discussed on page 9 of the *Installation Guide*.

A data set to be archived must be found on the volume at the time the request is made. Requests to process data sets on offline volumes are allowed only through sysparm TSOVOLOF.

The TSS command processors are described in detail in this section. TSS HELP text is available for all command processors.

DARCHIVE — Deferred Archive

A TSS command processor is provided to enable a user to queue an archive request for later batch processing. This command processor interfaces to the same subcontrol module used when a batch user makes a deferred archive request.

Successful use of this command results in the placement of a record containing all of the specified parameters into the ARCHCMDS file. All pending command records are processed and appropriately flagged at the next execution of the program that disposes of the deferred archive requests.

The status of such requests may be interrogated by use of the LISTREQ command after submission of the request. The processed command records are kept on file for an installation-specified number of days for possible inquiries. After this period has elapsed, they are automatically deleted. The default period is five days. Sysparms DARCKEEP and DRESKEEP may be specified to alter the default for either or both.

Restrictions

Deferred archiving does not support a subset of a PDS as specified by a member list.

DARCHIVE Command and Parameters

The command and parameters provided to request the deferred archival of a data set are described below:

```
DARCHIVE      DSN(data set name(s)) VOLUME(volser)
               NEWNAME(dsname) EXPDT(date) RETPD(nnnnn)
               UNCATALOG NOCATALOG RECATALOG SCRATCH NOSCRATCH
               RESET NORESET PASSWORD(password)
```

DSN

A list of one to ten names of data sets to be archived. The DSN can be eliminated from the command line and a prompt for a data-set name list will be issued. In the case of a GDG, either the relative or absolute name may be used.

VOLUME

This parameter must be supplied if either no entry exists in the system catalog for the data set, or if the catalog points to a data set with the same name on a different volume.

NEWNAME

A data set may be indexed in the archives under a different name, supplied as the value of this parameter. In the case of a GDG, either the relative or absolute name may be specified.

EXPDT

This parameter may be used to provide the date on which the archived data set may be deleted from the archives. It must be in an accepted SAMS:Disk format.

RETPD

An alternative means of specifying the date on which the archived data set may be deleted from the archives is to specify a number of days it is to be retained in the archives. The number of days must be a one- to four-digit number.

UNCATALOG

The normal default catalog action after a data set is archived, but it may also be specified.

NOCATALOG

This parameter may be used to preclude the default action of uncataloging at the completion of data set archival.

RECATALOG

The presence of this parameter results in the recataloging of a data set to the SAMS:Disk pseudo- volume upon successful archival.

SCRATCH

The normal default scratch action after a data set is archived, but it may also be specified.

NOSCRATCH

The scratching of a data set that normally occurs after archival is suppressed by the inclusion of this parameter.

RESET

By default, the change bit for the data set is reset (turned off) if the data set is NOT scratched. Although this is the default, it may still be specified.

NORESET

The resetting of the change bit that normally occurs when a data set is backed up (archived, but not scratched) is suppressed by the inclusion of this parameter.

PASSWORD

The password parameter provides the password value required to access a protected VSAM cluster. If the cluster is to be archived but not scratched, the read level or higher is required. If the cluster is to be scratched, the master-level password is required.

DRESTORE — Deferred Restore

A TSS command processor is provided to enable a user to queue a restore request for later batch processing. The same interface is used by the TSS foreground as the batch background services for processing deferred restore requests.

Execution of this command causes a record containing the specified parameters to be placed in the RESTCMDS file. These records are processed during the next batch execution of the program that disposes of the deferred restore requests.

At the time the queued restore requests are processed, they are sorted into sequence by the archive volumes on which the data sets reside, and block number sequence within volumes. This is done to minimize tape mounts and positioning time.

The RESTCMDS file may be interrogated by use of the LISTREQ command to determine if a request is still pending or if the restore has been completed. The queued request records will automatically be deleted five days (the default) after the restore processing has actually been done. This period to keep the records available for display can be changed by sysparm DRESKEEP.

DRESTORE Command and Parameters

The command and parameters provided to request the deferred restoring of a data set are described below:

```
DRESTORE      DSN(data set name(s)) NEWNAME(newname)
               VERSION(-nn) TIMEDATE(hhmmmyddd)
               VOLUME(volume) POOL(name) NOCATALOG RECATALOG
               SCRATCH ERASE PASSWORD(password)
```

DSN

A list of one to ten names of data sets to be restored. The DSN can be eliminated from the command line and a prompt for a data-set name list will be issued. The VERSION or TIMEDATE parameter may be used to designate other than the most recently archived version of a data set.

NEWNAME

To change the name of the data set being restored, this parameter must be specified.

VERSION

Specify -1 to -99 to restore the progressively older version of the data set to be restored. The absence of this parameter and the TIMEDATE parameter will default to restoring the most current version of the data set.

TIMEDATE

This parameter provides a more precise technique for designating a version of a data set to be restored. The format is hhmmmyydd and may be obtained from either the batch LISTD or TSS LISTDMS index listing for the data set.

VOLUME

This parameter may be specified to restore to a volume other than the original source volume.

POOL

To cause the data set to be restored to a volume belonging to a predefined POOL of volumes, this parameter is required.

NOCATALOG

The inclusion of this parameter prevents the data set from being cataloged after it has been successfully restored.

RECATALOG

Default processing attempts to catalog all data sets, but will not change an existing catalog entry. Specify this parameter only when you want to force the data set to be cataloged to the target volume.

SCRATCH

If the data set already exists on the target volume, this parameter causes it to be scratched and then reallocated.

ERASE

This parameter will cause the directory of a preallocated partitioned data set to be formatted before being used.

PASSWORD

The password parameter provides the password value required to access a protected VSAM cluster. If the cluster is to be restored but not scratched, the update-level or higher is required. If the cluster is to be scratched, the master-level password is required.

RESTORE — Immediate (Dynamic) Restore

Data sets can be dynamically restored in the TSS user region by way of the RESTORE command. A tape drive, if needed, is dynamically allocated by the command processor for the period of time necessary to restore the requested data set.

Note: Use of the dynamic (immediate) restore command requires that the RESTORE command processor be designated as privileged. This is necessary due to the use of the ENQ and ALLOCATE SVCs.

A facility is available to allow the installation to limit the number of tape units that can be concurrently allocated in support of TSS RESTORE commands. When the limit has been reached, execution of the command processor results in a response to the user that the command will have to be reissued at a later time.

This support does not require the dedication of tape units for its use. At the time of a restore request, an attempt is made to allocate a tape drive, if needed, according to the protocol observed by the system allocate function. Deallocation occurs immediately after the data set is restored.

RESTORE Command and Parameters

The command and parameters available to accomplish the dynamic restoring of data sets are specified as follows:

```
RESTORE      DSNNAME-LIST MEMBER(mbrlist) NEWNAME(newname)
              VERSION(-nn)
              TIMEDATE(hhmmyyddd)
              VOLUME(volume) POOL(name)
              NOCATALOG RECATALOG SCRATCH ERASE
              PASSWORD(password)
```

All parameter definitions for DRESTORE also apply to this command and are not repeated here. An additional capability in the immediate restore is to restore only selected members of a PDS. The members to restore are specified as follows:

```
MEMBER(mbrlist)
```

Mbrlist specifies one or more members to be restored from the PDS.

If a tape drive is needed and no tape unit is available for dynamic allocation at the time the command processor is executed, a message will be returned to the user and processing will terminate.

There is no restriction on the number of RESTORE commands that a user may execute during a session.

LISTDMS — Listing the Archive Index

A TSS command processor is provided to selectively list the contents of the archive index.

The command and parameters necessary to accomplish an inquiry are described below:

```
LISTDMS      DA(dsname-list) L(prefix-list) STA ALL ARC
              DUPLICATES
```

DATASET

A list of one or more data set names or patterns for which index entries are to be listed. The user ID will be prefixed to data set names not enclosed in quotes. DA('A.B.C') will list only the most current entry. DA('A.B.C') will list all occurrences.

Acceptable abbreviations for this keyword are DSN and DA.

LEVEL

Letter "L" is an acceptable abbreviation for level.

A list of levels or data set name patterns to be listed. As with other TSS commands, the user ID is not prefixed to level entries. A prefix does not have to coincide with an index level.

The entire index file can be listed by specifying L(/).

This keyword may be used instead of the DSNAME parameter.

STATUS

This default parameter may also be specified to cause data set status information to be listed. An alternative keyword is STA. This parameter is mutually exclusive with the ALLOCATION and ARCHIVES parameters.

ALLOCATION

The presence of this parameter causes data set allocation information to be displayed. An alternate keyword is ALL. This parameter is mutually exclusive with the STATUS and ARCHIVES parameters.

ARCHIVES

The presence of this parameter causes archive volume information for the data set to be listed. An alternate keyword is ARC. This parameter is mutually exclusive with the STATUS and ALLOCATION parameters.

DUPLICATES

This optional parameter indicates that all versions of the specified data sets are to be listed. An alternate keyword is DUP. If this parameter is not specified, only the most recent copy of the data set will be displayed.

If neither the data set nor the level parameter is specified, all entries under the user ID node will be listed. The status display is the default when no selection keyword has been specified.

LISTDMS — Sample Output**STATUS**

DSNAME	VOLSER	WHY	TIME	ARCDATE	EXPDATE
ADS.TEST.DATA1	ADS050	IMP	1713	1986.288	1988.348

ALLOCATION

DSNAME	DS	DEVICE	ALLOC	QTY	BLKS/MBRS
ADS.TEST.DATA1	PO	8595	TRK	15	50

ARCHIVES

DSNAME	ARCKEY	FILE	BLOCK	CNT	KBYTES
ADS.TEST.DATA1	TAPE01	269	6234	9	159

LISTREQ — List Status of Deferred Requests

A TSS command processor is available for displaying the status of requests for deferred archiving and restoring of data sets. The display will include the requests submitted in both TSS and batch environments.

The status indicated for each request will be displayed as one of the following values:

Table 16-1. LISTREQ Status Indicators and Descriptions

Indicator	Description
PEND	processing of the request is pending
COMP	processing of the request has completed successfully
FAIL	processing of the request has failed
ACTV	processing of the request is currently active

The request records will normally remain in their respective command files for five days subsequent to processing.

The command and parameters necessary to list the status of queued archive and restore requests are described below.

Note: Batch functions are also available for listing the status of deferred archive and restore requests. For detail information, see the *LISTREQ* command in the *BACKUP/ARCHIVE* and *RESTORE/RECOVER* sections, pages 205 and 240 respectively.

```
LISTREQ DA(dsname-list) L(prefix-list) ARCHIVE RESTORE
```

DATASET

A list of one or more data set names for which index entries are to be listed. The user ID will be prefixed to data set names not enclosed in quotes.

Acceptable abbreviations for this keyword are DSN and DA.

LEVEL

Letter “L” is an acceptable abbreviation for LEVEL.

A list of levels or prefixes of data set names to be listed. As with other TSS commands, the user ID is not prefixed to level entries. A prefix does not have to coincide with an index level.

The entire index file can be listed by specifying L(/).

This keyword may be used instead of the DSNNAME parameter.

ARCHIVE

The presence of this parameter results in queued archive request records being selected. It may be abbreviated as ARC.

RESTORE

The presence of this parameter results in queued restore request records being selected. It may be abbreviated as RES.

If neither the DATASET nor the LEVEL parameter is specified, all entries under the user ID will be listed.

If neither the ARCHIVE nor the RESTORE keyword is specified, RESTORE is assumed.

LISTREQ — SAMPLE OUTPUT

ARCHIVE / DSNNAME	HHMM	REQDT	STAT	HHMM	RUNDT
ADS.TEST.DATA2	1907	87031	PEND		
ADS.TEST.DATA1	1442	87028	COMP	1455	87028

DERASE — Erase a Deferred Request

The DERASE command has been provided to delete deferred archive and deferred restore requests.

Note: The DERASE command may also be used in batch runs. For detail information, see the *DERASE* command in the *BACKUP/ARCHIVE* and *RESTORE/RECOVER* sections, pages 206 and 241 respectively.

The command and parameters necessary to delete deferred requests are described below:

```
DERASE DSNAME(S) ARCHIVE RESTORE
```

DSNAME(S)

A list of one or more data set names for which deferred requests are to be deleted. A maximum of 20 data set names is allowed.

The user ID will be prefixed to data set names not enclosed in quotes.

ARCHIVE

The presence of this parameter indicates that the deferred requests to be deleted are archive transactions.

RESTORE

The presence of this parameter indicates that the deferred requests to be deleted are restore transactions.

If neither the ARCHIVE nor RESTORE keyword has been specified, RESTORE will be the default.

Restrictions

- Under VOS3, the use of the dynamic (immediate) restore command requires that the restore command processor be designated as privileged. Refer to the topic "*Activating TSS Support*" on page 25 in the *Installation Guide* for more information on how to authorize a command processor.
- The multitude of parameters unique to VSAM for altering cluster attributes are not supported in either the DRESTORE or the RESTORE command processors. Batch requests must be made whenever attributes are to be changed when restoring a cluster.

User Exits

User exits are provided that allow the installation to screen all TSS requests. Consult the section covering *User Exits*, beginning on page 195 in the *Systems Guide* for details regarding their use.

Chapter 17. Examples

The following examples of various SAMS:Disk executions are grouped by the SAMS:Disk function being performed.

REPORT Examples

REPORT Example 1

Produce the MULTIPLE VOLUME DATA SET DICTIONARY for all volumes beginning with the letters TSS.

```
//jobname JOB (acct,info),etc.
// EXEC DMS
//SYSIN DD *
SCAN REALVOLS
SELECT VOL=TSS/
REPORT MVD
```

REPORT Example 2

Another example illustrates the use of the DSNNAME keyword parameter where the Single Volume Dictionary is desired for all data sets beginning with the letters PERS on volume PRV002. The names of all the data sets appearing in the report are also to be saved in sequential data set LABS.MJB.DSNLIST

```
//jobname JOB (acct,info),etc.
// EXEC DMS
//SYSIN DD *
SCAN REALVOLS
SELECT VOL=PRV002,DSN=PERS/
REPORT SVD,CMDGENDSN=LABS.MJB.DSNLIST
```

Report Example 3

Scan all of the online volumes to find all proclib data sets, as identified by their data set names. Produce both the MVD report data for these data sets and a cross-reference of the procs that are duplicates (same names) in two or more of the libraries.

```
//jobname JOB (acct,info),etc.
// EXEC DMS
//SYSIN DD *
SCAN REALVOLS
SELECT VOL=/,DSN=(!PROCLIB!)
REPORT MVD,PO2,COMPONENTS
```

This list indicates that any data set name that ends with “PROCLIB” is to be processed. Since only non-VSAM data sets are being searched for, the COMPONENTS parameter was added to speed up processing.

REPORT Example 4

Compare the member names from MASTER.SOURCE and MASTER.LOADLIB and identify any member found in one library but not the other.

```
//jobname JOB (acct,info),etc.  
// EXEC DMS  
//SYSIN DD *  
    FIND DSN=MASTER.SOURCE  
    FIND DSN=MASTER.LOADLIB  
    REPORT PO1
```

or

```
//jobname JOB (acct,info),etc.  
// EXEC DMS  
//SYSIN DD *  
    FIND DSN=MASTER.SOURCE  
        REPORT PO1  
    FIND DSN=MASTER.LOADLIB  
        REPORT PO1
```

VREPORT Examples

VREPORT Example 1

Produce the VOLUME ALLOCATION SUMMARY for all volumes beginning with letters TSS.

```
//jobname JOB (acct,info),etc.  
// EXEC DMS  
//SYSIN DD *  
SCAN REALVOLS  
SELECT VOL=TSS/  
VREPORT ALL
```

ARCHIVE Examples

ARCHIVE Example 1

Archive data sets JAN.STATS and FEB.STATS. They should be scratched and uncataloged at the time of archival. The archive copies are to be kept for 60 days.

```
//jobname JOB (acct,info),etc.
// EXEC DMS
//SYSIN DD *
    FIND DSN=FEB.STATS
    FIND DSN=JAN.STATS
    ARCHIVE RETPD=60
```

Scratch and uncatalog are default actions. Note that these commands will result in a tape being mounted and the two named data sets being written to it (and indexed in the SAMS:Disk files).

ARCHIVE Example 2

Data sets JAN.STATS and FEB.STATS are to be archived, but they should be placed in the archive queue to be processed together with other archive requests. They are to be scratched and uncataloged at the time of archival. The archive copies are to be kept for 60 days.

```
//jobname JOB (acct,info),etc.
// EXEC ARCHIVE
//SYSIN DD *
    DARCHIVE DSN=JAN.STATS,RETPD=60
    DARCHIVE DSN=FEB.STATS,RETPD=60
```

Scratch and uncatalog are default actions. Note that these commands will result in no tape mount. The requests are merely placed in the archive queue in the SAMS:Disk files data set.

ARCHIVE Example 3

Create backup copies of DAILY.RECS and WEEKLY.SUMMARY. Place the requests in the queue to be processed with other requests. The backup copies need to be kept only 10 days. NOScratch and NOCatalog are both specified since backup copies are desired.

```
//jobname JOB (acct,info),etc.
// EXEC ARCHIVE
//SYSIN DD *
    DARCHIVE DSN=DAILY.RECS,NOS,NOC,RETPD=10
    DARCHIVE DSN=WEEKLY.SUMMARY,NOS,NOC,RETPD=10
```

ARCHIVE Example 4 - Dispose of Queued Requests

Process all of the requests that have been placed in the archive queue.

```
//jobname JOB (acct,info),etc.
// EXEC ARCHIVE,PARM.ARCHIVE=ADSDM276
```

No input stream is used, since the requests come from the archive queue in the SAMS:Disk files data set.

ARCHIVE Example 5 - List the ARCHIVE Request Queue

Find the current status for all archive requests that were made previously for data sets that begin with the name of DEPT20

```
//jobname JOB (acct,info),etc.
// EXEC LISTREQ
//SYSIN DD *
LISTREQ DSN=DEPT201/,ARCHIVE
```

A status of PEND (pending), COMP (complete) or FAIL (failed) will be displayed, along with other related data, for each archive/backup request found in the queue.

ARCHIVE Example 6 - Delete Entry from ARCHIVE Queue

An archive request was placed in the archive queue, but the wrong data set name was specified. Delete the entry from the queue.

```
//jobname JOB (acct,info),etc.
// EXEC DERASE
//SYSIN DD *
DERASE DSN=THE.WRONG.NAME,ARCHIVE
```

The ARCHIVE parameter instructs the delete to be made from the archive queue. (The same command, but specifying RESTORE instead of ARCHIVE, directs the delete to the restore queue.)

RESTORE Examples

RESTORE Example 1

Data set MUST.HAVE was accidentally destroyed. Restore the most current copy of the data set to the volume where the damaged copy now resides, deleting the damaged copy.

```
//jobname JOB (acct,info),etc.
// EXEC RESTORE
//SYSIN DD *
RESTORE DSN=MUST.HAVE,SCR
```

The SCRATCH parameter instructs SAMS:Disk to scratch the existing copy of the data set. By default, the data set will be restored to the volume to which it is cataloged, or if not cataloged, to its original volume. This request is not placed in the restore queue, but is executed immediately. To cause the request to be placed in the queue to be processed with other requests, simply change the command from RESTORE to DRESTORE.

RESTORE Example 2

Restore data set A.B. Do not use the most current backup version, but the previous version. Give the restored data set the name OLD.A.B, and restore it to any of the volumes defined to the pool of volumes known as WORKPOOL.

```
//jobname JOB (acct,info),etc.
// EXEC RESTORE
//SYSIN DD *
RESTORE DSN=A.B,VER=-1,NEW=OLD.A.B,POOL=WORKPOOL
```

The WORKPOOL of volumes must have been defined previously to SAMS:Disk. A sample definition for this pool and another named PRIVPOOL might be as follows. (The pool volumes will be used only if the new data set name is not currently cataloged. If it is, the cataloged volume will be used.)

Entries in DSN=SAMS.DISK.PARMLIB(DASDPOOL) to define the two pools:

```
'WORK01WORKPOOL'
'WORK02WORKPOOL'
'WORK03WORKPOOL'
'WORK04WORKPOOL'
'WORK05WORKPOOL'
'PRIV01PRIVPOOL'
'PRIV02PRIVPOOL'
```

RESTORE Example 3

Restore two VSAM clusters. The catalog being restored to is update password-protected, as is one of the clusters. Assign a new name to the other cluster and use the current date as its creation date. The cluster that is not being renamed is currently

defined in the catalog, so instruct SAMS:Disk to delete its old version prior to restore.

```
//jobname JOB (acct,info),etc.
// EXEC RESTORE
//SYSIN DD *
  PASSWORD CATALOG=VOL05 USERCAT,MASTER=UPDPASS
  RESTORE DSN=DMS.TJP.DMFTEST.CLUSTER,PASSWORD=DMFTEST,SCR
  RESTORE DSN=DMS.TJP.VSAM,NEWNAME=DMS.TJP.NEWVSAM,CREATE
```

RESTORE Example 4

Restore a VSAM cluster that was originally defined in a VSAM catalog into an DMF/EF Type-2 catalog. Change the volume to which it is being restored.

If the high level index in the master catalog has already been updated to point to the new DMFCAT1 catalog, all that is needed is the following command.

```
//jobname JOB (acct,info),etc.
// EXEC RESTORE
//SYSIN DD *
  RESTORE DSN=DMS.TJP.VSAM,VOLUME=VOL075
```

If the master catalog still relates the index to the VSAM catalog, you must supply the STEPCAT dd statement for the new catalog when you restore the cluster. Specifying the CATALOG= parameter on the RESTORE command is optional, since the presence of the STEPCAT will direct it there anyway.

```
//jobname JOB (acct,info),etc.
// EXEC RESTORE
//STEPCAT DD DISP=SHR,DSN=DMFCAT1
//SYSIN DD *
  RESTORE DSN=DMS.TJP.VSAM,VOLUME=VOL075,CATALOG=DMFCAT1
```

RESTORE Example 5

Restore a VSAM cluster and all of its alternate indexes. Rebuild the alternate indexes.

```
//jobname JOB (acct,info),etc.
// EXEC RESTORE
//SYSIN DD *
  RESTORE DSN=DMS.TJP.BASE.CLUSTER,AIXDEF,BLDINX
```

RESTORE Example 6

Restore a VSAM cluster and change its control interval size. Also, restore its two alternate indexes with different space allocations than they had originally.

```
//jobname JOB (acct,info),etc.
// EXEC RESTORE
//SYSIN DD *
RESTORE DSN=DMS.TJP.BASE.CLUSTER,CISZ=4096,
        NOAIXDEF      don't define the alternate indexes
RESTORE DSN=DMS.TJP.BASE.CLUSTER,
        AIXNAME=DMS.TJP.AIX.NAME1,          AIX's name
        ALLOC=(TRK,5,2)                    new space allocation
RESTORE DSN=DMS.TJP.BASE.CLUSTER,
        AIXNAME=DMS.TJP.AIX.NAME2,          AIX's name
        ALLOC=(CYL,2,0)                    new space allocation
```

RESTORE Example 7

Restore a base cluster that was archived with a master password of ORIGMSTR. Change the base cluster name and assign a new master password of NEWMSTR. Restore its alternate index with a new name also. Also change the data and index component names for the alternate index.

```
RESTORE      DSN=DMS.TJP.BASE.CLUSTER,
              PASSWORD=ORIGMSTR,          needed to read archive tape
              NEW=DMS.TJPNEW.BASE.CLUSTER, new base name
              MASTERPW=NEWMSTR,          new master password for base
              NOAIXDEF                    don't define the alternate index
RESTORE      DSN=DMS.TJP.BASE.CLUSTER,
              PASSWORD=ORIGMSTR,          needed to read archive tape
              RELATE=DMS.TJPNEW.BASE.CLUSTER, new base name
              RELPSWD=NEWMSTR,          new master password for base
              AIXNAME=DMS.TJP.AIX.NAME1,  AIX's original name
              NEW=DMS.TJPNEW.AIX.NAME1,    new AIX name
              DNAME=$.DATA,                new data component name
              INAME=$.INX                   new index component name
```

RESTORE Example 8 - Dispose of Queued Requests

Process all of the requests that have been placed in the restore queue.

```
//jobname JOB (acct,info),etc.
// EXEC RESTORE,PARM.RESTORE=ADSDM279
```

No input stream is used, since the requests come from the restore queue in the SAMS:Disk files data set.

RESTORE Example 9 - List the RESTORE Request Queue

Find the current status for all restore requests that were made previously for data sets that begin with the name of DEPT20

```
//jobname JOB (acct,info),etc.
// EXEC LISTREQ
//SYSIN DD *
LISTREQ DSN=DEPT201/,RESTORE
```

A status of PEND (pending), COMP (complete) or FAIL (failed) will be displayed, along with other related data, for each restore request found in the queue.

RESTORE Example 10 - Delete Entry from RESTORE Queue

A restore request was placed in the restore queue but the wrong data set name was specified. Delete the entry from the queue.

```
//jobname JOB (acct,info),etc.
// EXEC DERASE
//SYSIN DD *
DERASE DSN=THE.WRONG.NAME,RESTORE
```

The RESTORE parameter instructs the delete to be made from the restore queue. (The same command, but specifying ARCHIVE instead of RESTORE, directs the delete to the archive queue.)

Volume-Level BACKUP Examples

VBACKUP Example 1

Select volume "PROD01" for volume-level backup processing. Perform shared ENQs on all data sets in the volume but do not perform a RESERVE on the volume itself. Request a retention period of 30 days and create a MAP report of all tracks backed up in data set name sequence.

```
//jobname JOB (acct,info),etc.
// EXEC DMS
//SYSIN DD *
SCAN REALVOLS
SELECT VOL=(PROD01)
VBACKUP DSNENQ=SHR,RESERVE=NO,RETPD=00030,RPT=MAP
```

VBACKUP Example 2

Select volume "PROD01" for volume-level backup processing. Request no ENQ processing and put a RESERVE on the volume selected. Request an expiration date of 1999365 and do not create a MAP report. Also, backup only those tracks that are actually used, not all tracks that are allocated.

```
//jobname JOB (acct,info),etc.
// EXEC DMS
//SYSIN DD *
SCAN REALVOLS
SELECT VOL=(PROD01)
VBACKUP DSNENQ=NONE,RESERVE=YES,
EXPDT=1999365,RPT=NO,LSTAR
```

Volume-Level RECOVER Examples

VRECOVER Example 1

Selects volume "TSS006" for volume-level recovery processing. Perform no enqueues or reserve during volume-level recovery. After volume-level processing has completed successfully, all data set modified (or created) after the last volume-level backup are recovered using data set-level processing.

```
//jobname JOB (acct,info),etc.  
// EXEC RECOVER  
//SYSIN DD *  
VRECOVER VOL=TSS006,DSNENQ=NONE,RESERVE=NO,UPDATE
```

VRECOVER Example 2

Select volume "PROD01" for volume-level recovery processing. Recover the tracks starting with cylinder X'0115', head X'0005' and ending with cylinder X'0115', head X'001D'.

```
//jobname JOB (acct,info),etc.  
// EXEC RECOVER  
VRECOVER VOL=PROD01,ABS=(01150005-0115001D)
```

Volume Defragmentation Examples

Defragmentation Example 1 - Volume Defragmentation

Disk volumes can become fragmented to the degree that even though there are many free tracks available, requests to allocate new data sets will still fail. This is caused by the free space being in such small extents that it is unusable; that is, a track here, another there, but 15 tracks together can't be found.

Due to the lack of disk management tools in the past, the only recourse seemed to be to "reorganize" the pack, now commonly known as "defragging" it. Although there are times when this is needed, SAMS:Disk provides other alternatives that can solve the real problem with much less system impact.

The real problem is that space requests are failing. The term "fragmented" just explains why the requests are failing at a point in time. Why not just free up more space? A defrag operation usually means locking up the volume for several minutes, which often means running your entire machine in "stand-alone" mode as well. This is pretty severe just to gain some free space.

The simplest solution in many cases is to release the overallocated and unused space in the other data sets on the volume. This quite likely will actually increase the degree of free space fragmentation, but it may also provide much larger areas of contiguous free space, which solves the problem. This function does not require exclusive use of the volume (let alone the entire machine), and is very fast. If a volume defrag is still needed, having released the idle space beforehand provides an even greater return from the defrag run.

Another alternative is to move some of the data sets off to another volume and recatalog them. The Move/Copy function can do this with just one command. Data sets being used are skipped. The ones that are moved are also compressed, improving them as well as the target volume, and as they are scratched from the source volume, free space is made available. Again, exclusive control of only the data sets is needed, one at a time, not of the entire volume or machine.

Although the above options are highly recommended as first alternatives (and you may be able to think of several others), the point can still be reached when defragging the volume must be done. The following comments outline a method to do this with the ARCHIVE and RECOVER commands, as well as providing some added benefits and tips to keep in mind.

To SAMS:Disk, a volume defrag means the following:

1. Consolidate used tracks to obtain large contiguous free space extents.
2. Combine multiple extent data sets into a single extent. Compress PDS data sets.
3. Reorganize VSAM clusters.

4. Compress the VTOC.

A recommended approach to doing this with SAMS:Disk is as follows:

1. Allocate and format a new “mini” files data set.

The advantage of using a separate files data set for this special function is that it keeps these reorganization index entries separate from the archive/backup entries, and reduces the time needed to process the files data set, since it contains fewer entries.

2. Establish a JCL procedure that formats this files data set, and then use it in a DSCL run that archives and scratches all data sets on the pack. Follow this with a RECOVER command to restore all of the data sets on the volume.

The following JCL may be used as a guideline for “defragging” a disk volume.

First, create the files data set to be used.

```
//jobname JOB (acct,info),etc.
//INIT EXEC PGM=ADSMI002,PARM=ADSDM100
//STEPLIB DD DISP=SHR,DSN=SAMS.DISK.LOADLIB
//ABNLDUMP DD DUMMY
//CMDPRINT DD SYSOUT=A
//FILES DD DSN=SAMS.DISK.DEFRAG.volser.FILES,
// DISP=(,CATLG,DELETE),UNIT=SYSDA,
// DCB=(DSORG=DA),
// SPACE=(CYL,2,,CONTIG)
//MSGPRINT DD SYSOUT=A
//PARMLIB DD DISP=SHR,DSN=SAMS.DISK.PARMLIB
//SYSPRINT DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
```

Second, archive all non-VSAM data sets and VSAM clusters, from the desired volume using the special files data set.

```
//DEFRAGV EXEC DMS
//SYSIN DD *
//FILES DD DISP=SHR,DSN=SAMS.DISK.DEFRAG.volser.FILES
SCAN REALVOLS
SELECT VOL=volser
ARCHIVE
```

Third, recover all the data sets to the volume only if they were scratched successfully at archive time.

```
//RECOVER EXEC FMS
//FILES      DD DISP=SHR,DSN=SAMS.DISK.DEFRAG.volser.FILES
//SYSPARMS   DD *
PREALLOCN    no need to restore if not scratched after archive
VSPREDEFN    no need to restore if not scratched after archive
//SYSIN      DD *
SCAN DSNINDEX
SELECT DSN=dsnames
RECOVER TOVOL=volser
```

RECOVER Examples — FMS PROC

FMS — RECOVER Example 1

Recover between certain "date" criteria.

```
//jobname JOB (acct,info),etc.
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
  SELECT DSN=DISK820.BASE./,
    CRITERIA=(ARCDT,GT,TODAY-5,AND,ARCDT,LT,TODAY)
RECOVER POOL=MYPOOL,DISP=RECATALOG
```

FMS — RECOVER Example 2

Recover during different "time" criteria.

```
//jobname JOB (acct,info),etc.
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
  SELECT DSN=DISK820.BASE./,
    CRITERIA=(ARCDT,EQ,95201,AND,ARCTIME,GT,1200,
      AND,ARCTIME,LT,2100)
RECOVER POOL=MYPOOL,DISP=RECATALOG
```

FMS — RECOVER Example 3

Recover of archive data sets cataloged to the SAMS:Disk pseudo volume, allocating them onto volume DMS901, and altering each catalog entry from ARCIVE to DMS901.

```
//jobname JOB (acct,info),etc.
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
  SELECT DSN=DISK.XYZ./,
    CRITERIA=(ARCTYPE,EQ,RECAT)
RECOVER TOVOL=DMS901,DISP=RECATALOG
```

FMS — RECOVER Example 4

Recover data sets names contained in table FMSINCLD, as well as the data sets beginning with MYHLQ. Exclude data sets with an expiration date in the Format-1 DSCB if the date is less than 10 days from today.

```
//jobname JOB (acct,info),etc.
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
  SELECT TABLE=FMSINCLD,CRI=(ARCDT,NE,TODAY),VOL=(SSL/,SMS/)
  SELECT DSN=MYHLQ./
  EXCLUDE CRI=(EXPDT,LT,TODAY+10)
RECOVER
```

FMS — RECOVER Example 5

Recover data sets from the volume patterns SSL/ and SMS/ as long as they are located in table FMSINCLD, but not in table FMSEXCLD. Scratch any of the data sets if they already exist.

```
//jobname JOB (acct,info),etc.
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
  SELECT TABLE=FMSINCLD,VOL=(SSL/,SMS/)
  EXCLUDE TABLE=FMSEXCLD
RECOVER SCRATCH
```

FMS — RECOVER Example 6

Recover (and if necessary, scratch) all current backups of clusters located in an ARCHVOL named DISK820.DMSN.C1995306.T084345 and cataloging them into an alternate catalog.

```
//jobname JOB (acct,info),etc.
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
  SELECT TDSN=DISK820.DMSN.C1995306.T084345,
  CRI=(DSORG,EQ,AM,AND,ARCTYPE,EQ,BACKUP,AND,VERSION,EQ,0)
RECOVER SCRATCH,TOCAT=DMFCAT.BACKUP
```

FMS — RECOVER Example 7

Recover (and if necessary, scratch) all data sets from all the SSL8/ volume patterns, excluding volume SSL805. Also recover all catalogs except those backed up from the SMS/ volume pattern.

```
//jobname JOB (acct,info),etc.
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
  SELECT DSN=/,VOL=(SSL8/)
  EXCLUDE VOL=SSL805
  SELECT CAT=/,VOL=(/)
  EXCLUDE CAT=/,VOL=SMS/
RECOVER SCRATCH
```

FMS — RECOVER Example 8

Recover those data sets that were archived yesterday, excluding those data sets in FMSEXCLD. Any data set that happens to already exist will be left unaltered because sysparms PREALLOC and VSPREDEF both default to 'N'.

```
//jobname JOB (acct,info),etc.
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
  SELECT DSN=/,CRI=(ARCDT,EQ,TODAY-1,AND,
    ARCTYPE,EQ,ARCHIVE)
  EXCLUDE TABLE=FMSEXCLD
RECOVER
```

FMS — RECOVER Example 9

Recover non-VSAM data sets based on the following criteria:

- Begin with the pattern TSS
- Originally allocated in tracks
- The most current version
- Were archived and recataloged to the pseudo volume in the last 14 days.
- Exclude data sets contained in table FMSEXCLD
- Anything that was already restored today will be excluded

```
//jobname JOB (acct,info),etc.
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
  SELECT DSN=TSS/,CRI=(ALLOTTRKS,EQ,YES,AND,
    ARCDT,GT,TODAY-14,AND,VERSION,EQ,0,AND,
    ARCTYPE,EQ,RECAT,AND,DSORG,NE,AM)
  EXCLUDE TABLE=FMSEXCLD
  EXCLUDE CRI=(RESDT,EQ,TODAY)
RECOVER
```

FMS — RECOVER Example 10

Recover data sets based on the following criteria:

- non-VSAM
- Begin with the HLQ of ISPGHL1
- Were archived since yesterday at 08:43
- Exclude those data sets that have a F1EXPDT that is set to expire in the next 30 days
- Recover them to the HLQ of SBGLH

```
//jobname JOB (acct,info),etc.
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
  SELECT DSN=ISPGHL1./,CRI=(ARCDT,EQ,TODAY-1,AND,
    ARCTIME,GT,0843,AND,ARCTYPE,EQ,ARCHIVE,AND,DSORG,NE,AM)
  SELECT DSN=ISPGHL1./,
    CRI=(ARCDT,EQ,TODAY,AND,ARCTYPE,EQ,ARCHIVE,AND,DSORG,NE,AM)
  EXCLUDE CRI=(EXPDT,LE,TODAY+30)
RECOVER NEWHLQ=SBGLH
```

FMS — RECOVER Example 11

Recover (and if necessary, scratch) all data sets and catalogs archived to ARCHVOL @85347, excluding:

- Data sets listed in FMSEXCLD
- Catalog MASTER.CAT

```
//jobname JOB (acct,info),etc.
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
  SELECT DSN=/,CAT=/,CRI=(DSNKEY,EQ,<@85347>)
  EXCLUDE TABLE=FMSEXCLD
  EXCLUDE CAT=MASTER.CAT
RECOVER SCRATCH
```

FMS — RECOVER Example 12

During the execution of the following command, you received a tape I/O error on tape 123456 which caused the Recover to abend:

```
//jobname JOB (acct,info),etc.
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
  SELECT VOL=SSL805,
        CRI=(ARCTYPE,EQ,BACKUP,AND,VERSION,EQ,0)
RECOVER SCRATCH
```

Two items must be completed before restarting the Recover:

1. Exclude the ARCHVOL that caused the abend
2. Avoid recovering data sets that had previously been recovered in the first job

To do this, submit the following JCL:

```
//jobname JOB (acct,info),etc.
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
  SELECT VOL=SSL805,
        CRI=(ARCTYPE,EQ,BACKUP,AND,VERSION,EQ,0)
  EXCLUDE CRI=(DSNKEY,EQ,<123456>)
  EXCLUDE CRI=(RESDT,EQ,TODAY)
RECOVER SCRATCH
```

FMS — RECOVER Example 13

During the execution of the following command, an IPL occurred due to a system failure:

```
//jobname JOB (acct,info),etc.
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
  SELECT VOL=PROD/,CRI=(DSORG,EQ,AM,AND,ARCTYPE,EQ,BACKUP,
    AND,ARCDT,GT,TODAY-7),DSN=PROD./,VTOCS
  SELECT VOL=TSS/,CRI=(DSORG,NE,AM,AND,ARCTYPE,EQ,BACKUP,
    AND,ARCDT,GT,TODAY-7),DSN=TSS./,VTOCS
  SELECT TABLE=PAYROLL1,CRI=(DSORG,EQ,AM,AND,
    ARCTYPE,EQ,BACKUP,AND,ARCDT,GT,TODAY-7),VTOCS
  SELECT CAT=USER/,VOL=(PROD/,TSS/)
RECOVER SCRATCH
```

You need to restart the Recover as soon as the IPL is complete. Use the following JCL to restart this job and finish recovering the following criteria:

- The most current version of all VSAM clusters with the HLQ of PROD/, that were backed up from the PROD/ volume pattern in the last seven days, and were in the last VTOC backup
- The most current version of all non-VSAM data sets backed up with the HLQ of TSS/ from the TSS/ volume pattern, that were backed up in the last seven days, and were in the last VTOC backup
- The most current version of all VSAM clusters contained in the table PAYROLL1 that were backed up in the last seven days
- All catalogs starting with USER/ from volume patterns PROD/ and TSS/

```
//jobname JOB (acct,info),etc.
// EXEC FMS
//SYSIN DD *
SCAN DSNINDEX
  SELECT VOL=PROD/ ,CRI=(DSORG,EQ,AM,AND,ARCTYPE,EQ,BACKUP,
    AND,VERSION,EQ,0,AND,ARCDT,GT,TODAY-7) ,DSN=PROD./ ,VTOCS
  SELECT VOL=TSS/ ,CRI=(DSORG,NE,AM,AND,ARCTYPE,EQ,BACKUP,
    AND,VERSION,EQ,0,AND,ARCDT,GT,TODAY-7) ,DSN=TSS./ ,VTOCS
  SELECT TABLE=PAYROLL1 ,CRI=(DSORG,EQ,AM,AND,VERSION,EQ,0,
    AND,ARCTYPE,EQ,BACKUP,AND,ARCDT,GT,TODAY-7)
  SELECT CAT=USER/ ,VOL=(PROD/ ,TSS/)
  EXCLUDE CRI=(RESDT,EQ,TODAY)
RECOVER SCRATCH
```

BACKUP Examples

BACKUP Example 1

To illustrate the flexibility of DSCL BACKUP processing, some examples of typical use are provided.

The first example provides the command and entries necessary to accomplish the following:

- Archive, scratch and uncatalog all data sets on volume TEMPVS that are 14 days old.
- Scratch and uncatalog all data sets on all volume serials beginning with the letters DATA that are either uncataloged or have not been used in 30 days.
- Scratch and uncatalog all data sets on any volume being scanned that have not been used in 360 days.
- Test the actual run in simulation mode.

```
//jobname JOB (acct,info),etc.
// EXEC DMS
//SYSIN DD *
SET MODE=SIMULATE
SCAN REALVOLS
  SELECT VOL=TEMPVS,CRI=(USEDT,LT,TODAY-14)
  ARCHIVE
  SELECT VOL=DATA/,CRI=(CATSTAT,NE,YES)
  SELECT VOL=DATA/,CRI=(USEDT,LT,TODAY-30)
  SELECT CRI=(USEDT,LT,TODAY-360)
  DELETE
```

- The SET MODE=SIMULATE command is used to cause processing to be simulated. It may easily be removed after it has been verified that the desired results are being obtained.
- The VOL parameters are used to confine the processing to volume serials beginning with the letters specified.

BACKUP Example 2

This example is provided to illustrate the requirements for the following types of control.

- Scratch and uncatalog on all volumes that have serial numbers beginning with the letters "TSS" those data sets that are uncataloged and have not been used in 45 days.

- Archive, scratch, and uncatalog on all volumes that have serial numbers beginning with the letters USER all data sets that are uncataloged, have no expiration date, and have not been used in 90 days.

The DSCL commands required to accomplish the above is:

```
//jobname JOB (acct,info),etc.
// EXEC DMS
//SYSIN DD *
SET MODE=SIMULATE
SCAN REALVOLS
  SELECT CRI=(CATSTAT,NE,YES,AND,USEDT,LT,TODAY-45),
    VOL=TSS/
  DELETE
  SELECT VOL=USER/,CRI=(CATSTAT,NE,YES,AND,
    EXPDT,EQ,0,AND,
    USEDT,LT,TODAY-60)
  ARCHIVE
```

Incremental BACKUP Examples

Incremental BACKUP Example 1

To do incremental backup, one could use a daily job to capture all of the changed data sets, and a second weekly job to get backup copies of all the data that has not changed during the week. Full pack dumps are never needed again.

Monday Through Saturday Runs

```
//jobname JOB (acct,info),etc.  
// EXEC DMS  
//SYSIN DD *  
SCAN REALVOLS  
SELECT CRITERIA=(MODIFIED,EQ,YES)  
BACKUP RETPD=15
```

Sunday Run

```
//jobname JOB (acct,info),etc.  
// EXEC DMS  
//SYSIN DD *  
SCAN REALVOLS  
SELECT CRITERIA=(MODIFIED,EQ,YES)  
SELECT CRITERIA=(ARCDT,LE,TODAY-7)  
BACKUP RETPD=15
```

LISTD/LISTV Examples

LISTD/LISTV Example 1

List all of the archived data sets contained on tape DMS10

```
//jobname JOB (acct,info),etc.  
// EXEC LISTD  
//SYSIN DD *  
LISTD VOL=DMS105
```

LISTD/LISTV Example 2

List all of the archived data sets that contain either the name WOOD or BIRCH as the second-level qualifier.

```
//jobname JOB (acct,info),etc.  
// EXEC LISTD  
//SYSIN DD *  
LISTD DSN=(*.WOOD./,*.BIRCH./)
```

LISTD/LISTV Example 3

List all of the archive data sets, both from archive-to-disk and archive tapes.

```
//jobname JOB (acct,info),etc.  
// EXEC LISTD  
//SYSIN DD *  
LISTV
```

LISTD/LISTV Example 4

List all of archive tapes and archive-to-disk data sets that begin with the name of DMS.BACKUPS.

```
//jobname JOB (acct,info),etc.  
// EXEC LISTD  
//SYSIN DD *  
LISTV DSN=DMS.BACKUPS/
```

IXMAINT Examples

IXMAINT Example 1

Delete all of the expired data sets from the SAMS:Disk archives. If all of the data sets on any given archive volume (an archive tape or an archive data set on disk) have expired, delete the tape or disk data set from the archives as well.

```
//jobname JOB (acct,info),etc.  
// EXEC IXMAINT  
//SYSIN DD *  
DSNDELETE
```

IXMAINT Example 2

Delete archive tape DMS002 and all of the archived data sets contained on it, regardless of their expiration dates.

```
//jobname JOB (acct,info),etc.  
// EXEC IXMAINT  
//SYSIN DD *  
PURGE VOL=DMS002
```

MERGE Examples

MERGE Example 1

Merge the SAMS:Disk archive tapes (but not the backup tapes) as identified by names that start with DMS.ARCH. Merge only those tapes that are less than 50 percent full of unexpired data. Also generate a tape pull list for operations that can be used to find the needed tapes prior to actually starting the merge run. Create a duplicate copy of each new tape.

First, allocate the “pull-list” data set if not previously done.

```
//jobname JOB (acct,info),etc.
//ALLOC EXEC PGM=IEFBR14
//TAPEPULL DD DSN=SAMS.DISK.MERGE.TAPES,DISP=(NEW,CATLG),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120),
//          SPACE=(TRK,(1,1)),UNIT=SYSDA
```

To generate the tape pull list:

```
//TAPEMERG EXEC MERGE
//MERGE.TAPEPULL DD DISP=OLD,DSN=SAMS.DISK.MERGE.TAPES
//MERGE.SYSIN DD *
MERGE SIM,TDSN=DMS.ARCH/,PERCENT=50,TAPEPULL=GEN,
TYPE=TAPE
```

Note: that SIMULATE is used since generation of the pull list is the only processing required at this time.

To use the generated list of tapes and begin the actual merge: (System defaults will dynamically allocate one input drive and two output drives, one for the primary and one for the copy.)

```
//jobname JOB (acct,info),etc.
// EXEC MERGE
//MERGE.TAPEPULL DD DISP=OLD,DSN=SAMS.DISK.MERGE.TAPES
//MERGE.SYSIN DD *
MERGE TDSN=DMS.ARCH/,PERCENT=50,TAPEPULL=USE
```

MERGE Example 2

Merge the disk archives that are less than 90 percent full of unexpired data. Keep the primary copy of the merged output on disk, and a duplicate copy on tape.

```
//jobname JOB (acct,info),etc.
// EXEC MERGE
//MERGE.SYSIN DD *
MERGE SIM,TDSN=DMS.ARCH/,PERCENT=90,TYPE=DISK
//SYSPARMS DD *
MERPL1TYPDISK overrides the default of tape
MERC1TYPTAPE the default, but shown for illustration
```

MERGE Example 3

Merge some of the archives residing on disk volumes off to tape, making more room on disk for “new” archives. Select only the “old” archive data sets; that is, those that have been there for 60 days or more. Create a duplicate backup tape of each new archive tape.

```
//jobname JOB (acct,info),etc.  
// EXEC MERGE  
//MERGE.SYSIN DD *  
MERGE TYPE=DISK,DAYSOLD=60
```

REBUILD Examples

REBUILD Example 1

Rebuild the archive index entries for tape DMS544; that is, the index entry for the tape volume and the entries for all of the data sets contained on the tape. The name in the tape label for this tape is SAMS.DISK.ARCHIVES.G0032V00.

```
//jobname JOB (acct,info),etc.  
// EXEC REBUILD,DSN='SAMS.DISK.ARCHIVES.G0032V00',  
//          VOL=DMS544,UNIT=TAPE
```

REBUILD Example 2

Rebuild the archive index entries for the archive data set SYS.ARC.SAMS.DISK.D15APR85.T142003 that resides on disk volumes PACK01 and PACK14.

```
//jobname JOB (acct,info),etc.  
// EXEC REBUILD,DSN='SYS.ARC.SAMS.DISK.D15APR85.T142003',  
//          VOL='(PACK01,PACK14)',UNIT=SYSDA
```

IXUPDATE Examples

IXUPDATE Example 1 - DELETE

Delete all of the archive index entries for data set DEPT75.JOHN.TESTDAT1.

```
//jobname JOB (acct,info),etc.
// EXEC IXUPDATE
//SYSIN DD *
DELETE DSN=DEPT75.JOHN.TESTDAT1,ALL
```

IXUPDATE Example 2 - CHANGE

Update the expiration date for the current archive copy of DEPT75.JOHN from its present value to Jan. 1, 1984.

```
//jobname JOB (acct,info),etc.
// EXEC IXUPDATE
//SYSIN DD *
CHANGE DSN=DEPT75.JOHN,EXPDT=84001
```

IXUPDATE Example 3 - REQUEST

Since archive tape DMS879 was accidentally written over, disable its index entry such that all restore requests will automatically be directed to its duplicate copy tape.

```
//jobname JOB (acct,info),etc.
// EXEC IXUPDATE
//SYSIN DD *
RESET KEY=DMS879,DISABLE
```

IXUPDATE Example 4 - BUILDARC/RESET

Create an ARCHVOLS entry (entry for an archive tape) for tape DMS222 that is named DMS.ARCHCOPY. Then update the entry for DMS221 to indicate DMS222 as its duplicate copy tape.

```
//jobname JOB (acct,info),etc.
// EXEC IXUPDATE
//SYSIN DD *
BUILDARC VOL=DMS222,DSN=DMS.ARCHCOPY,TYPE=TAPE,ARCTYPE=COPY,
EXPDT=87031
RESET KEY=DMS221,COPY=DMS222
```

IXUPDATE Example 5 - RESET

Delete a copy volume from its primary volume.

```
//jobname JOB (acct,info),etc.
// EXEC IXUPDATE
//SYSIN DD *
RESET KEY=prim,COPY=NULL
RESET KEY=copy,SCRATCH
```

Idle Space Release Examples

RELEASE Example 1

Scan all real online volumes and select all PO data sets that are 50 percent overallocated. Release a portion of the space such that only 20 percent of the data set is idle.

```
//jobname JOB (acct,info),etc.
// EXEC DMS
//SYSIN DD *
SCAN REALVOLS
  SELECT CRITERIA=(DSORG,EQ,PO,AND,PCTUSED,LT,50)
  RELEASE PCTUSED=80
```

RELEASE Example 2

Scan all real online volumes and select all PS data sets that are 5 percent overallocated. Release all of the extra space.

```
//jobname JOB (acct,info),etc.
// EXEC DMS
//SYSIN DD *
SCAN REALVOLS
  SELECT CRITERIA=(DSORG,EQ,PS,AND,PCTUSED,LT,95)
  RELEASE
```

RELEASE Example 3

Release the idle space from those data sets that:

- Reside on real volumes starting with “TSS”.
- Have names that start with “ADS” or “PAY”.
- Are sequential, or are partitioned and have 20 percent or more idle space.

```
//jobname JOB (acct,info),etc.
// EXEC DMS
//SYSIN DD *
SCAN REALVOLS
  SELECT VOL=TSS/,DSN=(ADS/,PAY/),
         CRITERIA=(DSORG,EQ,PO,AND,PCTUSED,LT,80)
  SELECT VOL=TSS/,DSN=(ADS/,PAY/),
         CRITERIA=(DSORG,EQ,PS)
  RELEASE
```

Notice that all sequential data sets will have their idle space released, since no threshold is specified.

Sequential Migrate Examples

MIGRATE Example 1

Scan all real volumes that start with PLAN and migrate to tape all but the two most current generations of all GDG data sets.

```
//jobname JOB (acct,info),etc.  
// EXEC MIGRATE  
//SYSIN DD *  
MODESET SIMULATE  
SCAN VOL=PLAN/,GDGONLY,GEN=2
```

PDS Compression Examples

COMPRESS Example 1

Compress the data set named DEVE.TESTLIB, and all PDS data sets on the TSS volumes that are 80 percent full or over, or are in more than 15 extents. Reallocate the TSS PDSs such that they are only 50 percent full. Adjust the directory size such that the directory of each PDS will be 80 percent used.

```
//jobname JOB (acct,info),etc.
// EXEC COMPRES
//SYSIN DD *
COMPRESS DSN=DEVE.TESTLIB,USE
SCAN VOL=TSS/,THR=(80,15,00),ALL=(50,0,80),SHRINK
```

COMPRESS Example 2

Select from all WORK volumes only empty PDSs (that is, those having no members). Reallocate them using the minimum number of primary tracks or blocks that are necessary to store the original directory blocks.

```
//jobname JOB (acct,info),etc.
// EXEC COMPRES
//SYSIN DD *
SCAN VOL=WRK/,THR=(0,16),EMPTY,CONVERT
```

With empty PDSs that have a directory larger than one track, you may recover even more space by additionally specifying the SHRINK parameter. This will cause those directories to be reduced to a single track, permitting the minimum possible primary allocation of one track.

Move/Copy Examples

Move/Copy Example 1

Compress PDS PROD.TESTLIB by moving it to either PRIV01 or PRIV10.

```
//jobname JOB (acct,info),etc.
// EXEC DMS
//SYSIN DD *
  FIND DSN=PROD.TESTLIB
  MOVE TOV=(PRIV001,PRIV10)
```

Move/Copy Example 2

Convert a disk volume to a new device type. The recommended way to convert to new volumes and device types is to give the new volumes new names, and to then use the Move/Copy function to move and recatalog the data sets to the new volume.

```
//jobname JOB (acct,info),etc.
// EXEC DMS
//SYSIN DD *
  SCAN REALVOLS
  SELECT VOL=DA8595
  MOVE TOV=DA8598
```

If there is an overriding reason why you must give the new volume the same name as the old volume, it is recommended that you archive, scratch, and uncatalog the data sets from the source volume, then run volume recovery to the new volume that has been given the same name. When SAMS:Disk catalogs the data sets as they are being restored, the new device type will be put into the catalog entry.

```
//jobname JOB (acct,info),etc.
// EXEC DMS
//SYSIN DD *
  SCAN REALVOLS
  SELECT VOL=NVS001
  ARCHIVE
```

Clip (relabel) the target volume (the 8598) to the same volume serial as the source volume (the 8595). Vary the source volume offline. Mount the target volume on-line. Then execute the RECOVER command.

```
// EXEC RECOVER
//SYSIN DD *
  RECOVER VOL=NVS001,RECATALOG
```

DASD BILLING Examples

Set up DASD Billing for all volumes beginning with the letters WORK, and for volume LABS81. The billing rate for data sets on the WORK packs is \$ 008500 per kilobyte-day. For LABS81 it is \$ 010000 per kilobyte-day. The length of the billing cycle will be monthly and will begin on the first day of every month. Also, all transient data sets (those that are less than one day old on the volume) will not have charges accumulated for them.

First, allocate and initialize a separate files data set to be used for BILLING.

```
//jobname JOB (acct,info),etc.
//INIT      EXEC  PGM=ADSMI002,PARM=ADSDM100
//STEPLIB   DD  DISP=SHR,DSN=SAMS.DISK.LOADLIB
//ABNLDUMP  DD  DUMMY
//CMDPRINT  DD  SYSOUT=A
//FILES     DD  DSN=SAMS.DISK.BILLING.FILES,
//           DISP=(,CATLG,DELETE),UNIT=SYSDA,
//           DCB=(DSORG=DA),
//           SPACE=(CYL,10,,CONTIG)
//MSGPRINT  DD  SYSOUT=A
//PARMLIB   DD  DISP=SHR,DSN=SAMS.DISK.PARMLIB
//SYSPRINT  DD  SYSOUT=A
//SYSUDUMP  DD  SYSOUT=A
```

Second, establish the two parmlib members to define the volume categories and the billing rates for each.

```
DSN=SAMS.DISK.PARMLIB(DSBCATEG) contents:
    01WORK/X
    02LABS81
DSN=SAMS.DISK.PARMLIB(DSBRATEC) contents:
    01008500
    02010000
```

Third, accumulate the non-VSAM billing records for data sets on these volumes.

```
//jobname JOB (acct,info),etc.
// EXEC BILLING
//SELECT.SYSIN DD *
    SELECT VOLUME=(LABS81,WORK/)
//ACCUM.FILES DD  DSN=SAMS.DISK.BILLING.FILES,DISP=SHR
//ACCUM.SYSIN  DD *
    ACCUMULATE BMONTH,MODE=IMMEDIATE,BYPASSNEW
```

Fourth, accumulate the VSAM billing records for clusters defined on these volumes.

```
// EXEC VSAMBILL
//DMS.SYSIN DD *
SCAN CAT=/
  EXCLUDE CRITERIA=(DSORG,NE,VSAM)
  SELECT VOLUME=(LABS81,WORK/)
  BILLING
//ACCUM.FILES DD DSN=SAMS.DISK.BILLING.FILES,DISP=SHR
//ACCUM.SYSIN DD *
  ACCUMULATE BMONTH,MODE=IMMEDIATE,BYPASSNEW
```

After the billing period has passed (and several of the above accumulation jobs have been run), the billing totals are extended, printed, and cleared with the following JCL. This JCL should be executed on the last day of the billing cycle (if the cycle is monthly, it would be run on the 30th or the 31st day of the month, depending what day the month ends).

```
//jobname JOB (acct,info),etc.
// EXEC EXTEND
//EXTEND.FILES DD DSN=SAMS.DISK.BILLING.FILES,DISP=SHR
//SYSIN DD *
  EXTEND VOLUME=(LABS81,WORK/)
```

XCOPY Example

The XCOPY command shown below selects all archive volumes that have not already been processed by XCOPY, and creates a second set of archive and backup data sets. The new backup data sets are stored in an ARCHVOL created in the secondary files data set allocated to the OFILES DD statement.

```
//jobname JOB (acct,info),etc.
//COPYTAPE      EXEC XCOPY
//XCOPY.FILES   DD   DSN=DISK.PRIMARY.FILES,DISP=SHR
//XCOPY.OFILES   DD   DSN=DISK.OFFSITE.FILES,DISP=SHR
//XCOPY.SYSPARMS DD   *
MERP1NAMDMS.MERGPRI1
MERP1TYP3480
MERC1NAMDMS.MERGCOP1
MERC1TYPDISK
//XCOPY.SYSIN    DD   *
SELECT DSN=(PROD1./,PROD3./)
SELECT DSN=SYSA./
SELECT DSN=PRODACCT.MASTER.LOADLIB
SELECT DSN=CHKPOINT.UPDATE.LIB
XCOPY FLAG,CHKFLAG,TDSN=(APP1.BKUP/,APP2.BKUP/)
```

Glossary

Terms that are commonly used throughout data storage management, together with those terms unique to SAMS:Disk, are presented.

Archival

Provision for the storage of data at a lower cost, where it is not directly accessible to using programs. It includes the facility for restoring the data to the original medium on a demand basis.

Archvol(s)

The tape volume(s) or disk data set(s) that contain the data sets that have been archived. Also the index to them that is kept in the files data set.

Auto-Restore

A SAMS:Disk process by which data sets that have been archived are restored automatically if they are required by an application at a later time. The data set may be restored by an online job or a batch job. This allows the DASD manager to keep free space available, yet gives the user access to data sets that are referenced infrequently. Data sets that have been archived to disk rather than to tape can be auto-restored with no operator intervention.

Backup

A process by which a copy of a DASD data set is made, usually on tape, for the purpose of allowing the data set to be restored in the event of loss or damage to the original data set. Backup differs from the archive process, in that the data sets are copied, but not scratched or uncataloged.

SAMS:Disk backup processing provides for incremental backup, which limits copying to those data sets that have been changed since the last backup was taken. As each changed data set is identified, the backup copy is created and indexed, and then the change bit is turned off. To protect the data sets that are not changed, they are backed up regularly, such as once a week.

See also “*Full Pack Backup*” on page 461.

Compress

The term “compress” has three different meanings in SAMS:Disk.

- In PDS Compress, the term “compress” describes the process of eliminating “dead space” in a PDS, or shrinking the size of the data set or its directory, by writing the PDS to a workfile, calculating reallocation requirements, then copying it back.

- The second usage applies to archive or backup processing. When data sets are copied to an archive medium, only active and used data blocks are read, thus resulting in PDS and VSAM data sets being copied to the archive medium in compressed format. Subsequent restore or volume recovery results in the consolidation of several extents into one, a process sometimes referred to as “defragmentation,” and also the compression/reorganization of the data set.
- The third usage also applies to archive or backup processing, and is sometimes referred to as “compaction”. When data is being copied to an archive media, the data is analyzed and encoded (encrypted) into a new format that occupies less space. When the data set is restored, the data is automatically expanded (or de-compressed) into its original format.

Data Set index — DSNINDEX

A file containing a record for each data set that has been archived and has not yet expired. Separate records are created for each version of a data set that has been archived more than once. The existence of the data set index makes it possible to restore an archived data set by referencing only its name.

Deferred processing

A means by which requests for certain actions are recorded and saved for later processing (queued processing). This is usually done to allow common requests to be grouped and processed in bulk in a more efficient manner, or to provide greater integrity by ensuring that certain actions are complete before other actions are begun.

Dispose processing

The actual execution of those requests that at some prior time were deferred (queued); i.e., deferred actions are disposed of at a later time.

DSCB — Data Set control block

A record maintained by the operating system in a special DASD data set for each data set stored on a DASD volume. It contains attributes such as record format, block size and record length, as well as information about space occupied.

DSCL — Data storage command language

A set of SAMS:Disk commands providing a comprehensive, efficient common means to process VSAM and non-VSAM data sets. These commands are preprocessed by SAMS:Disk to group all overlapping requests to allow all actions to be performed during a single pass of VTOCs or catalogs.

Explicit Processing

Any SAMS:Disk processing in which the data set(s) to be processed are not found by scanning either VTOCs or catalogs, but by processing a list of explicit names

that either the user has provided, or have been extracted from the catalog. See “Implicit Processing”.

File (first use)

A data set that has been archived and now exists as one of numerous internal files to the larger archive data set. Such a file (representing a single archived data set) must be restored by a SAMS:Disk utility function before it can be referenced by an application program.

File (second use)

A set of logically related records maintained by SAMS:Disk as a subfile of a single BDAM data set. The entire BDAM data set is referred to as the “files data set”, while each subfile is usually referenced by its specific file name, such as the “data set index file”. Each file (subfile) is defined by an entry in the FILEDEFN member of parmlib.

Full Pack Backup

SAMS:Disk provides two different methods to back up a full DASD volume. The first method, full pack backup, processes data at the data set level, and is invoked by the RETAIN procedure, using the F-type selection criteria in a volspec entry. This method allows you to restore individual data sets explicitly using the RESTORE command, or recover a group of data sets using the RECOVER command. Turn to page 165 for a further description.

The second method uses the SAMS:Disk Volume-level backup method and is invoked by the VBACKUP command, making use of DSCL. Data is restored by using the VRECOVER command. Turn to page 251 for a further description.

Implicit Processing

Any SAMS:Disk processing in which the data set(s) selected for (or excluded from) processing are determined by scanning the input source, such as VTOCs or catalogs, from beginning to end, and applying user-specified criteria to determine what is or is not to be processed. The criteria may include factors such as last-use date, expiration date, create date, catalog status, blocksize, or being listed in a table. SAMS:Disk permits multiple selection or exclusion criteria to be specified.

Merge

The redistribution of archived data sets residing within several archive data sets (on tape or disk) to a smaller number of new archive data sets. Expired data sets are deleted, and the unexpired ones are sorted and grouped to use the archive media more efficiently.

Migration

The movement of data from one type of storage medium to another without restricting accessibility to using programs. Reasons for migration include obtaining a lower-cost storage medium and changing the grouping of data.

PARMLIB — Parameter Library

A standard partitioned data set containing both system-supplied members and user-supplied members that control various aspects of SAMS:Disk processing.

Prefix or Group

A set of data sets or members that is identified with a common name prefix. A group name is indicated by placement of a slash (/) immediately following the last common character in a data set or member name. All data sets or members that have names that match on the number of characters preceded by the slash (/) are considered part of that group.

Restore

The process by which an individual data set is restored (i.e., copied from a backup or archive copy) to a DASD device. SAMS:Disk permits restore processing without preallocation for VSAM, PS, PO and DA data sets as well as for OS CVOL catalogs. Individual members of a PDS may also be restored. See also “Recover”.

Recover

The process by which a group of data sets, an entire volume, or a group of volumes is recovered (i.e., copied from a backup or archive copy) to DASD. SAMS:Disk recover processing provides for the same capabilities as restore processing, plus permits additional controls to be specified for the greater flexibility necessary for volume recovery.

Retention Control

An automated means by which SAMS:Disk determines which data sets should be retained on disk and which removed.

Stand—Alone Restore

Stand—Alone Restore is a SAMS:Disk program (known as DMSSAR), provided as a Selectable Unit, that is designed to restore an entire DASD volume without the presence of the VOS3 operating system. It runs under the Hitachi/M-series, Hitachi/ES and Hitachi/SA instruction sets and runs under VMS as a guest operating system. The Stand—Alone Restore Selectable Unit is documented in the *Stand—Alone Restore Guide*.

In order to provide the ability to perform a stand—alone restore, SAMS:Disk has developed volume-level processing (as opposed to its normal data set-level processing mode) for backup and recover functions. These are documented beginning on page 251.

SYSPARM — System Parameter

A means by which SAMS:Disk can be tailored and implemented to meet the unique processing needs of each installation. Sysparms control various aspects of each SAMS:Disk function. The default values are in effect unless they are overridden either temporarily (through overriding DD statements) or permanently (through specification in the SYSPARM member of the parmlib data set).

Systems Managed Storage

The ability of an installation to specify the storage management rules and methodologies needed for their unique installation, and to then automate these within the system itself. Systems Managed Storage not only removes most if not all of the clerical effort and intervention that has historically been required, but must also provide the installation with control and flexibility in specifying its own rules and methodologies. If this second quality is overlooked, Systems Mismanaged Storage can easily be the result. The ability to manage and understand the automated system is of equal if not greater importance.

Volume — Level Processing

SAMS:Disk normally processes at the data set level. However, in order to provide the capability to perform Stand-Alone Restore, SAMS:Disk can perform backup and recover processing at the volume level. These functions are documented beginning on page 251.

The volume-level backup function (VBACKUP) is invoked through DSCL. It will produce a backup tape or tapes containing an entire DASD volume, starting at cylinder 0, track 0 and continuing through all used cylinders.

The volume-level recover function is invoked under the RECOVER procedure, but through the VRECOVER command. When specified at recovery time, it overlays the entire volume in extent track order (i.e., from cylinder 0 to the end of the used cylinders).

VTOC — Volume Table of Contents

The data set on each DASD volume that contains a record called a Data Set Control Block (DSCB) for each data set stored on that volume. It also carries records that are used to account for free space.

Index

!

6951 MTL 17

A

Abend

 Merge Abend/Restart Considerations 319
 User Abend Codes 11

Abnormalities Detected 15

Absolute Addressing

 BDAM Considerations 337

ACCESS 187, 192

Access Methods for VSAM Clusters 174

Action

 Command, DSCL COPY 345
 Command, DSCL MOVE 342
 DSCL RELEASE Action Command 356

Actions

 Catalog Actions 204
 Delete Actions 204
 Reset Change Bit Actions 205

Additional

 Copies of Archive Tapes, Making 310
 Parameters, PDS Compression 368

Addressing

 Absolute, BDAM Considerations 337
 Relative Block, BDAM Considerations 336
 TTR, BDAM Considerations 336

Aging, Rapid, at Midnight, Problems with 168

AIXS, VSAM

 RESTORE for, Without Defer 229

ALL Parameter 307

Allocated in blocks (yes/no) 55, 89

Allocated in cylinders 55, 89

Allocated in tracks 90

Allocated in tracks (yes/no) 56

Allocation 417

 Data Set Allocation 12

 Space, Restore 211

 Target 332

Alternate Index

 and Path New Names, Determination of 218

 Processing Clusters with 173

 Restore 218

Restoring	219
Support for	175
Volume Selection for	219
AMDCINV	56 - 57
AMDNCIS	58
AMDSTSP	51, 61
AND/OR Logic, DSCL Select Criteria	68
AND/OR Logic, FMS Select Criteria	95
Anomaly	
PDS	171, 199
Archival	
SYSCTLG	201
Versus Sequential Migration	374
ARCHIVE	
ARCDT	50
ARCHIVE	206
Back Up or Archive Empty Clusters	172
BACKUP and ARCHIVE Considerations	168
Command, DSCL	190
Deferred	190, 201
Deferred (DARCHIVE)	200
Deferred, DARCHIVE	411
Delete Entry from Queue, Example	427
DSCL Command Examples	193
Erasing of Deferred ARCHIVE Requests (DERASE)	206
Example, Dispose of Queued Request	427
Example, List the Request Queue	427
Examples	426
EXPLICIT	199, 202
EXPLICIT ARCHIVE and DARCHIVE Command Parameters	203
Explicit Condition Codes	201
Explicit, MODESET Command Parameters	202
Index Maintenance	280 - 283
Index Rebuild Utility	304
Index, List (LISTDMS)	416
List Deferred Requests (LISTREQ)	205
List Indexes	283
List Status of Deferred Requests (LISTREQ)	418
Management Considerations	254
Merge of Archive Data Sets	313
Non-VSAM BACKUP and ARCHIVE Considerations	171
Processing Queued ARCHIVE Requests (Dispose)	205
Purging Archive Data Sets/Volumes	302
Report Sequencing Options	170, 200
Restrictions	190, 201
Syntax	191
Tapes, Making Additional Copies of	310
VSAM BACKUP and ARCHIVE Considerations	172
VSAM Backup and Archive Restrictions	174

ARCHIVE, ARCHIVE date	90
ARCHIVE, ARCHIVE time	90
ARCHIVES	417
ARCHVOL Key	91
Assignment	
of Creation Dates to Output Data Sets	376
Attributes of Data Sets, Change	215
Authority	
PASSWORD, Password-protected (yes/no)	62
Auto-Restore	314, 328, 374, 459
Recataloging for VSAM Auto-Restore	174

B

BACKUP	
ARCDT	50
Back Up or Archive Empty Clusters	172
BACKUP and ARCHIVE Considerations	168
Command Examples, DSCL	188
Deferred	185
Deferred Requests	185
DMF/EF Type-2 Catalog Backup and Recovery	176
DSCL Command	185
Examples	443
Incremental	180
Incremental, Examples	445
Incremental, Implementing	180
Incremental, Restrictions	184
of SYSCTLG	171
of Volume VTOCS	169
Restrictions	185
Syntax	186
Volume-Level Examples	432
VSAM BACKUP and ARCHIVE Considerations	172
VSAM Backup and Archive Restrictions	174
BACKUP, BACKUP date	90
BACKUP, BACKUP time	90
BACKUP/ARCHIVE date	90
BACKUP/ARCHIVE time	90
BDAM Considerations	336
Absolute Addressing	337
RECFM=U or V	337
Relative Block Addressing	336
TTR Addressing	336
Billing	
See DASD Billing	385
Bit	
Change Bit	168
Reset Change Bit Actions	205

Block size (number)	56
Boolean Logic, DSCL Select Criteria	68
Boolean Logic, FMS Select Criteria	95
BUILDARC, Volume Index Command	308
BUILDARC/RESET	
IXUPDATE, Example	451

C

Catalog	
Catalog Actions	204
Default Delete, Catalog, and Change Bit Resetting Actions . . .	200
Default Processing	214
DMF/EF Type-2 Catalog Backup and Recovery	176
DMF/EF Type-2 Catalog Recovery	221
OS CVOL (SYSCTLG), Backup of	171
PASSWORD, RESTORE	224
Catalog Concerns, Sequential Migration	382
Catalog status of the data set	57
CATALOG=	83, 85
Change	
Change Bit	168
Command, TIMEDATE Parameter	307
Data Sets Attributes	215
Default Delete, Catalog, and Change Bit Resetting Actions . . .	200
DSNINDEX EXPDT Command	307
IXUPDATE, Example	451
Reset Change Bit Actions	205
CIAOPT	187, 192
Clusters	
Access Methods for	174
Empty	215
Empty, Back Up or Archive	172
HURBA	215
in Use by Other Tasks, Processing	173, 216
Multivolume, Volume Selection for	220
Non-Standard Formats, Processing	178
Optional Data Report at Restore Time	217
Predefined and Reusable, Restoring to	217
Processing Considerations	348
Restore and Recover of	213 - 224
Restore and Recover of, Restrictions	213
RESTORE Command Parameters for	228
RESTORE for, Without Defer	230
SAMS:Disk Verify of VSAM Clusters	172
with Alternate Indexes, Processing	173
CMA-SPOOL	15
Codes	
Explicit ARCHIVE Condition Codes	201

for Tape Pull List	240
User Abend Codes	11
Command	
DSCL ARCHIVE Examples	193
MERGE Command Parameters Description	325
Commands	26, 78
Action, DSCL COPY	345
Action, DSCL MOVE	342
Action, DSCL RELEASE	356
ARCHIVE, DSCL	190
BILLING, DSCL	73
BUILDARC	308
Delete Data Set Index Command	306
DELETE, DSCL	73
DSCL BACKUP	185
DSCL BACKUP, Examples	188
DSCL DELETE	195
DSCL DELETE Examples	196
DSCL EXCLUDE COMPONENTS	70
DSCL EXPIRE	197
DSCL EXPIRE Examples	197
DSCL Sequence	27
DSNDELETE Command Parameters Description	298
EXCLUDE CATALOG	97
EXCLUDE CRITERIA	97
EXCLUDE DSNAMES	70, 96
EXCLUDE TABLE	96
EXCLUDE TDSN	96
EXCLUDE VOLUMES	96
EXCLUDE VTOCS	97
EXCLUDE, DSCL	70
EXCLUDE, FMS	96
EXPLICIT ARCHIVE and DARCHIVE Command Parameters	203
Explicit ARCHIVE MODESET Command Parameters	202
FIND, DSCL	41
FMS Sequence	79
LISTD	286
LISTF	294
LISTV	284
MERGE	325
MERGE, JCL	323 - 324
MIGRATE, Explicit Migration	378
MODESET Parameter Descriptions	378
Parameters, DARCHIVE	411
Parameters, DRESTORE	413
Parameters, RESTORE	415
PASSWORD for RESTORE	224
PASSWORD, DSCL (For VSAM Only)	36
PASSWORD, FMS (For VSAM Only)	83

PURGE	302
REBUILD	305
RECATALOG, Parameter Descriptions	383
RECOVER AIX= - FMS PROC	246
RECOVER CREATE - FMS PROC	246
RECOVER DEFALIAS= - FMS PROC	247
RECOVER DISP= - FMS PROC	245
RECOVER ERASE - FMS PROC	246
RECOVER EXPORTF= - FMS PROC	247
RECOVER NEWHLQ= - FMS PROC	246
RECOVER NOLOAD - FMS PROC	246
RECOVER POOL= - FMS PROC	245
RECOVER SCRATCH - FMS PROC	246
RECOVER TOCAT= - FMS PROC	247
RECOVER TOVOLUME= - FMS PROC	245
RECOVER, FMS	98, 244
RELEASE Examples	356
RELEASE, DSCL	73
RELOAD	369
RESTORE	225
RESTORE Command Parameters for VSAM Clusters	228
SCAN CATALOGS	39
SCAN REALVOLS, DSCL	38
SCAN, DSCL	38
SCAN, FMS	84
SCAN, Implicit	362
SCAN, Implicit Migration to Tape	379
SELECT, DSCL	44
SELECT, FMS	85
SET, DSCL	35
SET, FMS	81
VBACKUP, DSCL	73
VOLDELETE Command Parameters Description	301
XCOPY	263
Components	
Determination of Data and Index Component New Names	217
VSAM Data, RESTORE for, Without Defer	234
VSAM Index, RESTORE for, Without Defer	237
Compression	
and RESTORE/RECOVER	15
Data Set Compression and Reorganization	170, 199
Examples	454
Compression, PDS	
See Also, PDS Compression	359
Condition Codes	297, 306
DASD Billing	396
DSCL	25
Explicit ARCHIVE	201
FMS	78

for Tape Pull List	240
LISTD	283
MERGE	324
PDS Compression	361
RECOVER - FMS PROC	245
Restore	208
Sequential Migration to Tape	377
Settings	297
Considerations, Special	13
Control Area splits (number)	57
Control Interval splits (number)	58
Control Statements, Basic Format	4
Control, Retention	
See, Retention Control	165
Copies, Duplicate, of Migration Tapes	382
COPY	
DSCL Action Command	345
Copy command	331
ENQ/DEQ	332
Examples	455
General Information	332
Messages	334
Restrictions	334
Sample Commands	350
Sample Report	351
Scratch/Catalog Defaults	332
Target Allocation	332
User Exits	341
COPY Parameter	307
Creation date	50, 91
Creation Dates	
for Output Data Sets, Assignment of	376
CRITERIA	
ALLOBLKS, Allocated in blocks (yes/no)	55
ALLOCYLS, Allocated in cylinders	55
ALLOPRI, Primary allocation amount (number)	56, 89
ALLOSEC, Secondary allocation amount (number)	56
ALLOTRKS, Allocated in tracks (yes/no)	56
ALOBLKS, Allocated in blocks (yes/no)	89
ALOCYLS, Allocated in cylinders	89
ALOTRKS, Allocated in tracks	90
ARCDT	45
ARCDT Last BACKUP/ARCHIVE date (date format)	50
ARCDT, BACKUP/ARCHIVE date	90
ARCTIME, BACKUP/ARCHIVE time	90
ARCTYPE, Type of DSNINDEX record	90
BLKSZ	105 - 106
BLKSZ, Block size (number)	56
CASPLITS, Control Area splits (number)	57

CATSTAT, Catalog status of the data set	57
CISPLITS, Control Interval splits (number)	58
CREDIT, Creation date	91
CREDIT, Creation date (date format)	50
Data Set Selection, Sequential Migration	373
DSNKEY, ARCHVOL Key	91
DSORG, Data set organization	58, 91
EXPDT, Expiration date	92
EXPDT, Expiration date (date format)	51
Extents in the data set (number)	58
F1CREDIT, Format-1 DSCB create date	53
F1EXPDT, Format-1 DSCB expiration date	55
F1MODDT, Format-1 DSCB last modified date	54
F1USEDT, Format-1 DSCB last used date	54
Field Name Descriptions	50
GDG, Generation data group generation (y/n)	59
GDGGEN, Relative generation number (number)	59
GDGSTAT, Generation data group status	59
IDLETRKS, Idle tracks (number)	60
JOBNAME, in the Format-1 prior to archival	92
KBYTES, Kilobytes allocated to the data set	60, 92
Logic	68
Logic, FMS	95
LRECL, Logical record length (number)	60
MODBIT, Modification (change) bit (on/off)	60
MODDT, Last Modification date	92
MODDT, Last modification date (date format)	51
MODIFIED	45
MODIFIED, Modified since last backup	61
PASSTYPE, Password protection type	62
PASSWORD, Password-protected (yes/no)	62
PCTUSED, Percent used (number)	62
PDSORG, Partitioned data set organization	63
PSORG	64
RACFIND, RACF-indicated (yes/no)	64
RECAT, Psuedo Volume used when archived	93
RECFM, Record format	64
RESDATE, last RESTORE date in DSNINDEX record	93
RESTIME, last RESTORE time in DSNINDEX record	93
SVCMODE, SVC mode field for VTOC statistics	65
TEMP, System temporary non-VSAM data set	65
TODAY, Today's date (date format)	52, 94
TRKS, Tracks allocated	65
UNMOVE, Unmovable attribute (yes/no)	66
USEDT, Last Use date	94
USEDT, Last Use date (date format)	52
USERFLD, User Field Information	94
VERSION, Version # of the DSNINDEX record	94
VERSIONA, ARCHIVE Version of DSNINDEX records	95

VERSIONA, BACKUP Version of DSNINDEX records	95
VOLCNT, Data set allocated volume count	66
VOLSEQNO, Volume sequence number	66
VSAMORG, VSAM organization type	67
CRITERIA=	86

D

DA and PS Data Set	14
DA Data Set Attribute Conflict	14
DARCHIVE	
Command Parameters	411
Deferred Archive	411
EXPLICIT ARCHIVE and DARCHIVE Command Parameters	203
Restrictions	411
DASD Billing	385
Accumulate JCL for VSAM Clusters	400
Condition Codes	396
Examples	456
Implementation, Suggested Run Procedures	402
Processing Considerations	402
Data Set Names, Exclude from Migration	333
Data Set Organization	91
Data Sets	
Merge of Archive Data Sets	313
Types SAMS:Disk Can Process	13
Data Storage Command Language	
See, DSCL	24
Date	197
ARCDT Last BACKUP/ARCHIVE date	50
Creation, for Output Data Sets, Assignment of	376
Expiration, Data Sets on Tape	374
Last Use, Maintenance	215
Parameter	284
Date Format	8
Gregorian	8
Julian	9
Default	
Catalog Processing	214
Delete, Catalog, and Change Bit Reset	200
Scratch/Catalog	332
DEFER	
and Restore	227
DEFER	187, 192, 205
RESTORE for Non-VSAM, Without	227
RESTORE for VSAM AIXS, Without	229
RESTORE for VSAM Clusters, Without	230
RESTORE for VSAM Data Components, Without	234
RESTORE for VSAM Index Components, Without	237

RESTORE for VSAM, Without	228
DEFERRED	
Archive (DARCHIVE)	411
Archive Request, List (LISTREQ)	205
Deferred Archive	201
Deferred Archives	190
Deferred Archives (Queued Archive Requests)	200
Deferred Backup	185
Deferred Backup Requests	185
Erase Deferred ARCHIVE Requests (DERASE)	206
GDGSTAT, Generation data group status	59
List Status of Requests (LISTREQ)	418
Request, Erase from the Queue (DERASE)	241, 420
Restore	210
Restore, DRESTORE	413
Restore, List of Requests (LISTREQ)	240
Defragmentation	
Examples	434
Volume, and Implicit Recovery	242
Volume, Example	434
Volume, Secondary Extent Consolidation	214
Defragmentation, Volume	248
DELETE	
Default Delete, Catalog, and Change Bit Resetting Actions	200
Delete Actions	204
Delete Data Set Index Command	306
DSCL Command	195
DSCL Command Examples	196
Entry from ARCHIVE Queue, Example	427
Entry from the RESTORE Queue, Example	431
Expired Data Sets	297
Expired Volumes	299
IXUPDATE, Example	451
Overriding Erase Option During	216
Unexpired Data Sets	173, 215
Using NOSCRATCH Option During	216
DELPROF Parameter	301
DERASE	
Erase a Deferred Request from the Queue	241, 420
Erasing of Deferred ARCHIVE Requests	206
Descriptions	
DSNDELETE Command Parameters Description	298
LISTD Report	293
LISTF Report	295
MERGE Command Parameters Description	325
MODESET Command Parameter	378
PURGE Command Parameters Descriptions	302
RECATALOG Command Parameter	383
VOLDELETE Command Parameters	301

Determination	
of Alternate Index and Path New Names	218
of Data and Index Component New Names	217
Device	
Independence and Ease of Use	199
DISABLE	308
DISP	191, 195
Dispose	
Dispose of Queued Request, ARCHIVE Example	427
Processing Queued ARCHIVE Requests	205
DMF/EF Type-2	
Backup and Recovery	176
Catalog Recovery	221
DMSAR	174, 191, 193 - 195, 314, 328, 374, 459
DRESTORE	
Command Parameters	413
Deferred Restore	413
DS1AVR	
Allocated in blocks (yes/no)	55
DS1CREDIT	50, 53
DS1CYL	55 - 56
DS1DSGU	66
DS1EXPDT	51, 55
DS1IND02	60 - 61
DS1IND10	62
DS1IND40	64
DS1LRECL	60
DS1LSTAR	
IDLETRKS, Idle tracks (number)	60
PCTUSED, Percent used (number)	62
DS1NOEPV	58, 61
DS1SCAL3	61
DS1SVCMO	65
SVCMODE	55
DS1TRK	55
DS1VOLSQ	66
DSCB	
Model DSCB Processing	171, 336
MODELDSCL, Is the data set a model DSCB?	61
DSCL	24
ARCHIVE Command	190
BACKUP Command	185
BACKUP Command Examples	188
Command Sequence	27
Commands	26
Condition Codes	25
COPY Action Command	345
DELETE Command	195
DELETE Command Examples	196

DSCL RELEASE Action Command	356
EXCLUDE Command	70
Exclude COMPONENTS	70
Exclude DSNAMES	70
EXPIRE Command	197
EXPIRE Command Examples	197
FIND Command	41
Logic	68
Messages	26
MOVE Action Command	342
PASSWORD Command	36
REPORT Command	102
Sample DSCL REPORT Commands	140
SCAN and FIND Rules	28
SCAN CATALOGS Command	39
SCAN Commands	38
SCAN REALVOLS Command	38
SELECT and EXCLUDE Rules	31
SELECT Command	44
SELECT CRITERIA Field Name Descriptions	50
SET Command	35
DSCLSCEX	341
DSETEXDT	50 - 51
DSNAME	203, 205 - 206
DSNAMES=	85
DSNDELETE	
Command Parameters Description	298
DELPREF Parameter	301
DSNINDEX	84
Change DSNINDEX EXPDT Command	307
Record Update Userid Field, LISTF	295
DSORG	91
DSORG, Data set organization	58
Duplicate Copies	
of Migration Tapes	382
Duplicate Tape	
Recatalog to, JCL	383
DUPLICATES Parameter	
LISTD Command	286
LISTF Command	294
Dynamic	
Allocation	11
Restore	415

E

Ease of Use and Device Independence	199
Empty Clusters	215
Back Up or Archive	172
ENQ/DEQ	332
Enqueue, User Data Set Integrity	18
Erase	
a Deferred Request from the Queue (DERASE)	241, 420
Deferred ARCHIVE Requests (DERASE)	206
Overriding During Delete	216
Error Recovery, VSAM	349
ESF Spool data sets	15
Examples	423
ARCHIVE command	30 - 31
BACKUP command	30, 33
DELETE command	34
DSCL Commands	74
DSCL REPORT Commands	140
FIND DSNAMEs	30
RECOVER Command - FMS PROC	437
REPORT command	30
REPORT MVDICT	29, 31
SCAN CATALOG	30 - 31
SCAN REALVOLS	31 - 33
SELECT CATALOG	33
SELECT CRITERIA=(ARCDT...	30
SELECT CRITERIA=(CATSTAT...	57
SELECT CRITERIA=(CREDIT...	31
SELECT CRITERIA=(DSORG...	31
SELECT CRITERIA=(EXTENTS...	44
SELECT CRITERIA=(GDGGEN...	59
SELECT CRITERIA=(MODDT...	30
SELECT CRITERIA=(PASSTYPE...	62
SELECT CRITERIA=(PCTUSED...	30
SELECT CRITERIA=(PDSORG...	63 - 64
SELECT CRITERIA=(RECFM...	65
SELECT CRITERIA=(RECFMB...	65
SELECT CRITERIA=(TRKS...	30
SELECT CRITERIA=(VOLSEQNO...	66
SELECT CRITERIA=(VSAMORG...	67
SELECT VOLUMES	32
SELECT VTOCS	33
EXCLUDE	
Data Set Names from Migration	333
DSCL EXCLUDE Command	70
DSCL EXCLUDE COMPONENTS	70
DSCL EXCLUDE DSNAMEs	70
DSCL Rules	31

FMS EXCLUDE CATALOG	97
FMS EXCLUDE CRITERIA	97
FMS EXCLUDE DSNAMES	96
FMS EXCLUDE TABLE	96
FMS EXCLUDE TDSN	96
FMS EXCLUDE VOLUMES	96
FMS EXCLUDE VTOCS	97
Exempting	
Data Sets	168
Restored Data Sets	168
EXPDT	92, 186, 191, 203
Command, Change DSNINDEX EXPDT	307
Parameter	308
RESET Command	308
Expiration Date	51, 92
Tape Data Sets, Sequential Migration	374
EXPIRE	
DSCL Command	197
DSCL Command Examples	197
Expired	
Data Sets, Deleting	297
Volumes, Deleting	299
EXPLICIT	
ARCHIVE	199
ARCHIVE Command Password Parameter	205
ARCHIVE Condition Codes	201
ARCHIVE MODESET Command Parameters	202
COMPRESS Commands	367
EXPLICIT ARCHIVE and DARCHIVE Command Parameters	203
MIGRATE	378
Explicit Archive	
GDG	202
Extent, Secondary, Consolidation	214
Extents in the data set (number)	58

F

Field Descriptions	
LISTD Report	293
LISTF Report	295
Files Data Set	
Multiple, for Recover	244
Multiple, for Restore	208
Updates for RESTORE/RECOVER	222
Files data set maintenance	279
FIND, DSCL Rules	28
Flags	
LISTD Flags	291
LISTV Flags	284

FMS	
Command Sequence	79
Commands	78
Condition Codes	78
Exclude Catalog	97
EXCLUDE Command	96
Exclude Criteria	97
Exclude DSNAMES	96
Exclude Table	96
Exclude TDSN	96
Exclude Volumes	96
Exclude VTOCS	97
Logic	95
PASSWORD Command	83
RECOVER AIX=	246
RECOVER Command	244
RECOVER CREATE	246
RECOVER DEFALIAS=	247
RECOVER DISP=	245
RECOVER ERASE	246
RECOVER EXPORTF=	247
RECOVER NEWHLQ=	246
RECOVER NOLOAD	246
RECOVER POOL=	245
RECOVER SCRATCH	246
RECOVER TOCAT=	247
RECOVER TOVOLUME=	245
SCAN Command	84
SELECT Command	85
SET Command	81
FMS PROC	78
Examples	437
Format	
Date	8
Gregorian Date	8
Julian Date	9
Non-Standard, Processing Clusters with	178
Format-1 DSCB	
create date	53
expiration date	55
last modified date	54
last used date	54

G

GDG	202
GDGGEN, Relative generation number (number)	59
GDGSTAT, Generation data group status	59
Generation data group generation (yes/no)	59
Restore Considerations for GDG Data Sets	212
GEN= Parameter	81
Grace Periods, Restore	210
Gregorian Date Format	8

I

Idle Space Release	353
Examples	452
General Information	354
Messages	354
SYSPARMS	355
IDLETRKS, Idle tracks (number)	60
Immediate Restore	415
Implementation	
Merge Implementation Concerns	316
Implicit	
Recovery and Volume Defragmentation	242
SCAN Command	362
SCAN Command, Migration to Tape	379
Incremental Backup	180
Examples	445
General Information	180
Implementing	180
Monday Through Saturday Runs	445
Restrictions	184
Sunday Run	445
Independence, Device, and Ease of Use	199
Index	
and Data Component New Name Determination	217
Archive, List	283
Archive, List (LISTDMS)	416
BUILDARC Command	308
Delete Data Set Index Command	306
Listings, JCL	283
Rebuild, JCL	305
Updates, JCL	306
Volume and Data Set Index Updates (IXUPDATE)	306
Index, Alternate	
and Path New Names, Determination of	218
Processing Clusters with	173
Restore	218
Restoring	219

Support for	175
Volume Selection for	219
Index, VSAM Components	
RESTORE for, Without Defer	237
Installation	
Options, Sequential Migration to Tape	372
Integrity	
Data Considerations	254
Data Set, User	18
IXMAINT	
Examples	447
JCL	297
Utility	296
IXUPDATE	
CHANGE, Example	451
DELETE, Example	451
Examples	451
REQUEST, Example	451
RESET, Example	451
Volume and Data Set Index Updates (IXUPDATE)	306
IXUPDATE, BUILDARC/RESET, Example	451

J

JCL	
Index Listings	283
Index Updates	306
IXMAINT	297
JCL for Processing Queued Restore Requests	239
MERGE Command	323 - 324
PDS Compression	362
Rebuild Index	305
Recatalog to Duplicate Tape	383
Requirements	11
RESTORE	223
Sequential Migration to Tape	378
JOBNAME	92
Julian Date Format	9

K

KBYTES, Kilobytes allocated to the data set 60, 92

L

Last modification date 51, 92

Last RESTORE Date in DSNINDEX record 93

Last RESTORE Time in DSNINDEX record 93

Last Use date 94

Last Use Date Maintenance 215

Last Used Block Pointer 256

Legend

REPORT Fields 109, 116, 118, 120, 125

List

Archive Index (LISTDMS) 416

Deferred Archive Request (LISTREQ) 205

Index, JCL 283

List Archive Indexes 283

LISTREQ 205

of Deferred Restore Requests (LISTREQ) 240

PDS Member 199

RESTORE Request Queue, Example 431

Status of Deferred Requests (LISTREQ) 418

Tape Pull, Condition Codes for 240

Tape Pull, for Queued Restore Requests 240

the ARCHIVE Request Queue, Example 427

LISTD

Command 286

LISTD Flags 291

Report Field Description 293

Sample LISTD Report 295

LISTD/LISTV

Examples 446

LISTDMS

List the Archive Index 416

Sample Output 417

LISTF

Command 294

Report, Field Descriptions 295

LISTREQ

List of Deferred Restore Requests 240

List Status of Deferred Requests 418

Sample Output 419

LISTV

Command 284

DATE Parameter 284

LISTV Flags 284

Logic, Merge Processing	321 - 322
LRECL	60

M

Maintenance, Files data set	279
Maintenance, Last Use Date	215
Management	
ARCHIVE Management Considerations	254
MASTER	
PASSWORD, RESTORE	224
MASTER=	83
MEMBER	187, 193, 203
PDS Member Restore	211
Member Level Archive	190
Member Level Backup	185
Member List	
List, PDS	199
Memory Requirements	11
Merge	
Command	325
Command Parameters Description	325
Concerns, Operation	316
Condition Codes	324
Examples	448
JCL	323 - 324
Merge Abend/Restart Considerations	319
Merge Implementation Concerns	316
Merge of Archive Data Sets	313
Processing Logic	321 - 322
Rationale Overview	314
Sample Reports	329
Specifying Output Options	315
Specifying What to Merge	314
Uncatalog Parameter	328
Messages	11, 354
DSCL	26
Move/Copy commands	334
See also SAMS:Disk Message Manual	
Methods, Access, for VSAM Clusters	174
Midnight Rapid Aging Problems	50, 168
MIGRATE	
Command, Explicit Migration	378
Commands	378
Migrating Protected Data Sets	376
Migration	
Concerns	382
Data Set Names, Exclude from	333
Implicit, to Tape, SCAN Command	379

Sample Report	384
Tapes, Duplicate Copies of	382
Migration to Tape, JCL	378
MODBIT	
MODBIT, Modification (change) bit (on/off)	60
UNMOVE, Unmovable attribute (yes/no)	66
MODDT, Last modification date	51
MODE= Parameter	81
Model DSCB	61
Processing	171, 336
MODELDC, Is the data set a model DSCB?	61
MODESET	
Explicit ARCHIVE MODESET Command Parameters	202
MODESET Command Parameter Descriptions	378
MODIFIED since last backup (yes/no)	61
MOVE	
DSCL Action Command	342
Move command	331
ENQ/DEQ	332
Examples	455
General Information	332
Messages	334
Restrictions	334
Sample Commands	350
Sample Report	351
Scratch/Catalog Defaults	332
Target Allocation	332
User Exits	341
MSGLEVEL=	82
Multiple Files Data Set for Recover	244
Multiple Files Data Set for Restore	208
Multivolume	
Clusters, Volume Selection for	220
Data Sets	17
General Information	100
Restore	209
Tape Processing	374
VOLCNT, Data set allocated volume count	66
VOLSEQNO, Volume sequence number	66

N

Names

Determination of Data and Index Component New Names . . .	217
NEWHLQ	187, 193
NEWNAME	187, 192, 203
NOCATALOG Parameter	204, 227
NOEXTNT	58
Non-VSAM	
BACKUP and ARCHIVE Considerations	171
Data Sets, Restore Parameters for	227
NORESET	187, 205
NOSCRATCH	204
Using During Delete	216

O

Optional Cluster Data Report at Restore Time	217
Options	
ARCHIVE Report Sequencing	170
Archive Report Sequencing Options	200
Installation, Sequential Migration to Tape	372
Overriding Erase Option During Delete	216
Special Processing	214
Specifying Merge Output	315
Speed	214
Using NOSCRATCH Option During Delete	216
OS CVOL Catalog (SYSCTLG), Backup of	171
Output	
Sample (LISTDMS)	417
Specifying Merge Options	315
Output Data Sets	
Assignment of Creation Dates to	376
Overflow Data Sets, Track	253
Overriding Erase Option During Delete	216

P

Parameters

Additional Parameters	368
ALL	307
ALLOCS (VREPORT Command)	107
ATTRBS (VREPORT Command)	108
BACKUP (SCAN Command)	39
CATALOG (FIND Command)	42
CATALOG= (PASSWORD Command)	83
CATALOG= (SELECT Command)	85
CATALOGS (EXCLUDE Command)	72
CATALOGS (SCAN Command)	39

CATALOGS (SELECT Command)	46
CIAOPT	103
CIARPT (REPORT Command)	103
CLDATA (REPORT Command)	102
CLDOPT	102
CLOSE (SMF Report)	138
CMDGENDSN (REPORT Command)	106
Command, DRESTORE	413
COMPONENTS (REPORT Command)	107
COMPONENTS (SELECT Command)	45
COPY	307
COUNT (SMF Report)	138
CRITERIA (EXCLUDE Command)	72
CRITERIA (SELECT Command)	47
CRITERIA= (SELECT Command)	86
DARCHIVE	411
DATE	284
DELPROF	301
Descriptions, RECATALOG Command	383
DISTRs (VREPORT Command)	108
DSCL VBACKUP Commands and Parameters	255
DSINDX (REPORT Command)	104
DSN (SMF Report)	138
DSNAME (FIND Command)	42
DSNAMES (SELECT Command)	44
DSNAMES= (SELECT Command)	85
DSNDELETE Command Parameters Description	298
DSNINDEX (SCAN Command)	84
DSUTIL (REPORT Command)	104
EXCCATS (SCAN Command)	39
EXCLUDE (REPORT Command)	107
EXCLUDE COMPONENTS	70
EXPLICIT ARCHIVE and DARCHIVE Command Parameters	203
Explicit ARCHIVE MODESET Command Parameters	202
FLAGDAYS	107
FREESP	108
GEN= (SET Command)	81
LAYOUT (VREPORT Command)	108
MAPPER (REPORT Command)	105
MASTER= (PASSWORD Command)	83
MEMBER (REPORT Command)	105
MEMBERS (REPORT Command)	107
MERGE Command Parameters Description	325
MODE= (SET Command)	81
MODESET Command Descriptions	378
MODULES (REPORT Command)	107
MSGLEVEL= (SET Command)	82
MVDICT (REPORT Command)	105
NOCATALOG	227

NOLOAD	227
PASSWORD (FIND Command)	42
PDSTAT (REPORT Command)	105
PO1REF (REPORT Command)	106
PO2REF (REPORT Command)	106
POCCHH (REPORT Command)	105
PODUMP (REPORT Command)	105
POOL	227
POTTRD (REPORT Command)	106
POXREF (REPORT Command)	106
PURGE Command Parameters Descriptions	302
RECATALOG	227
RESTORE	415
RESTORE Command Parameters for VSAM Clusters	228
RESTORE for Non-VSAM Without Defer	227
RESTORE for VSAM AIXS Without Defer	229
RESTORE for VSAM Clusters Without Defer	230
RESTORE for VSAM Data Components, no Defer	234
RESTORE for VSAM Index Components, no Defer	237
RESTORE for VSAM Without Defer	228
RPT	260
Scan, Sequential Migration	373
Select, Sequential Migration	373
SHOWVOL (SCAN Command)	38
SMF report command and parameters	138
START (SMF Report)	138
STOP (SMF Report)	138
SVDICT (REPORT Command)	106
SYSCTLG	227
TABLE (EXCLUDE Command)	71
TABLE (SELECT Command)	45
TABLE= (SELECT Command)	86
TAPE (SMF Report)	139
TDSN= (SELECT Command)	86
TIMEDATE	307
UNIT	305
VOLDELETE Command Parameters Description	301
VOLUMES (EXCLUDE Command)	71
VOLUMES (SELECT Command)	46
VOLUMES= (SELECT Command)	86
VRECOVER Commands and Parameters	258
VTOCMSG= (SET Command)	81
VTOCS (EXCLUDE Command)	71
VTOCS (SELECT Command)	46, 86
PASSTYPE, Password protection type	62
PASSWORD	205
Command, Restore	224
PASSTYPE, Password protection type	62
Password Detail Group (P)	114

Password-protected (yes/no)	62
Password-Protected Data Sets	16
SCAN Command	367
Path New Names	
and Alternate Index, Determination of	218
Patterns and Data Set Names	5
PDS	
Anomalies	171, 199
Member Level Archive	190
Member Level Backup	185
Member List	199
PDS Member Restore	211
PDS Compression	359
Commands, Explicit	367
Condition Codes	361
Examples	454
General Information	360
JCL	362
Sample Report	370
PDSORG, Partitioned data set organization	63
Percent used (number)	62
Performance Considerations	
Bypassing VVDS and catalog overhead	107
DSCL Select Command	44
POTTRD REPORT	106
REPORTS	101
Performance, Task	169
Period	
Grace, for Restore	210
POOL	
Parameter	227
Preallocated Data Sets, Restore to	211
Predefined and Reusable Clusters, Restore to	217
Prefix	
Volume	8
Primary allocation amount (number)	56, 89
PRIMSPAC	56
Problems	
Rapid Aging	50
Problems, Midnight Rapid Aging	168
Processing	
Clusters in Use by Other Tasks	173, 216
Clusters that have Non-Standard Formats	178
Clusters with Alternate Indexes	173
Considerations for VSAM Clusters	348
Default Catalog	214
JCL for Processing Queued Restore Requests	239
Model DSCB	171, 336
Multivolume Tape	374

Queued ARCHIVE Requests (Dispose)	205
Special Options	214
SYSCTLG	337
TAPEPULL	317
Volume-Level, Overview of	252
Protected Data Sets, Migrating	376
PS and DA Data Sets	14
PS Data Set Attribute Conflict	14
PSE Data Sets	64
PSORG	64
Psuedo Volume used when archived	93
PURGE	
Command Parameters Descriptions	302
Purge Archive Data Sets/Volumes	302

Q

Queue

Erase a Deferred Request from the, (DERASE)	420
Erase a Deferred Request from, (DERASE)	241
Processing Queued ARCHIVE Requests (Dispose)	205
Queued Archive Requests	190, 200
Queued Backup Requests	185
Tape Pull List for Restore Requests	240

Queued

Dispose of Request, ARCHIVE Example	427
JCL for Processing Queued Restore Requests	239
Restore Requests	210

R

RACFIND, RACF-indicated (yes/no)	64
Rapid Aging at Midnight, Problems with	168
Reblocking	339
Reblocking Specifications	339
Rebuild	
Archive Index Rebuild Utility	304
Command, UNIT Parameter	305
Examples	450
Index, JCL	305
REBUILD Command	305
RECAT	93
RECATALOG	
Command Parameter Descriptions	383
Parameter	204
Recataloging for VSAM Auto-Restore	174
to Duplicate Tape, JCL	383
RECATALOG Parameter	227
RECFM	64

U or V, BDAM Considerations	337
RECOVER	
and Restore of VSAM Clusters	213 - 224
and Restore of VSAM Clusters, Restrictions	213
Condition Codes - FMS PROC	245
Empty Clusters	215
FMS PROC	244
FMS RECOVER AIX=	246
FMS RECOVER CREATE	246
FMS RECOVER DEFALIAS=	247
FMS RECOVER DISP=	245
FMS RECOVER ERASE	246
FMS RECOVER EXPORTF=	247
FMS RECOVER NEWHLQ=	246
FMS RECOVER NOLOAD	246
FMS RECOVER POOL=	245
FMS RECOVER SCRATCH	246
FMS RECOVER TOCAT=	247
FMS RECOVER TOVOLUMES=	245
Multiple Files Data Set for	244
Restore Considerations for GDG data sets	212
Volume-Level Examples	433
RECOVER - FMS PROC	
Examples	437
Recovery	
DMF/EF Type-2 Catalog Backup and Recovery	176
DMF/EF Type-2 Catalog Recovery	221
Implicit, and Volume Defragmentation	242
VSAM Error	349
Relative Block Addressing	
BDAM Considerations	336
Release	
Command Examples	356
DSCL Action Command	356
Idle Space, SYSPARMS	355
Sample Report	358
RELEASE Command, CONVERT	356
RELEASE Sysparm, RLSEUNMV	355
RELOAD Command	369
Renaming Non-VSAM Data Sets	333
Renaming VSAM Data Sets	333
Reorganization	
Data Set Compression and Reorganization	199
Reorganization and Data Set Compression	170
REPORT	
ALLOCS (See VREPORT)	107
Analyzing the CIA Report	135
Archive Report Sequencing Options	200
ARCHIVE Sequencing Options	170

ATTRBS (See VREPORT)	108
Bypassing VVDS and catalog overhead	107
CIA Report Description	116
CIARPT, Cluster Internal Analysis	103
CLDATA Report Description	111
CLDATA, Cluster Data Analysis	102
Copy command	351
DISTRS (See VREPORT)	108
DSCL REPORT Command	102
DSINDX Data Set Index Report	104
DSUTIL Data Set Utilization Report	104
Excluding member names	107
Field Legend	109, 116, 118, 120, 125
FREESP (See VREPORT)	108
Identifying unused data sets	107
Including member names	107
LAYOUT (See VREPORT)	108
LISTD Report Field Description	293
LISTF Field Descriptions	295
MAPPER, Volume Map	105
MEMBER, PDS Member Directory	105
Merge Data Set Volume Status	329
MIGRATION	384
Move command	351
MVDICT, Multiple Volume Data Set Dictionary	105
Optional Cluster Data Report at Restore Time	217
Passing control to a module	107
PDS Compress, Sample	370
PDSTAT, Partitioned Data Set Status	105
PO1REF, DSN X-Ref for Member in a Single PDS	106
PO2REF, DSN X-Ref for Member in Multiple PDSs	106
POCCHH, PDS Directory (CCHHR Sequence)	105
PODUMP, PDS Directory (Member Sequence)	105
POTTRD, PDS Directory (TTR Sequence)	106
POXREF, DSN X-Ref for PDS Member Names	106
Printing options for CIARPT	103
Printing options for CLDRPT	102
Restrictions	101
Sample DSCL REPORT Commands	140
Sample LISTD Report	295
Sample RELEASE Report	358
Sample Restore/Recover Report	249
Saving data set names selected for reporting	106
Sequential Migration to Tape	377
SMF Report, JCL	138
SVDICT, Single Volume Data Set Dictionary	106
VBACKUP Example	261
REPORT Examples	423
VREPORT	425

Report LISTD	295
Request	
Deferred, Erase from the Queue (DERASE)	241, 420
Erasing of Deferred ARCHIVE Requests (DERASE)	206
IXUPDATE, Example	451
List Deferred Archive Requests (LISTREQ)	205
List the ARCHIVE Queue, Example	427
Processing Queued ARCHIVE Requests (Dispose)	205
Queued Archive	190, 200
Queued Backup	185
Tape Pull List for Queued Restore Requests	240
Requests	
Deferred, List Status of (LISTREQ)	418
JCL for Processing Queued Restore Requests	239
Sequencing Restore Requests	208
Requirements	
and Basic Conventions	4
JCL	11
Storage	11
RESDATE	93
Reset	307
Command, COPY Parameter	307
IXUPDATE, Example	451
RESET	205
Reset Change Bit Actions	205
Reset Actions	
Default Delete, Catalog, and Change Bit Resetting Actions	200
Restart	
Task Restart Considerations	170
RESTIME	93
Restore	
Alternate Indexes	218
and Recover of VSAM Clusters	213 - 224
and Recover of VSAM Clusters, Restrictions	213
Command Parameters	225, 415
Command, Inowritechk Parameter	239
Condition Codes	208
Date and Time	222
Deferred	210
Deferred, DRESTORE	413
Delete Entry from the Queue, Example	431
Empty Clusters	215
Examples	428
Grace Periods	210
Immediate	415
JCL	223
JCL for Processing Queued Restore Requests	239
List of Deferred Requests (LISTREQ)	240
List Status of Deferred Requests (LISTREQ)	418

List the Request Queue, Example	431
Multiple Files Data Set for	208
Multivolume	209
NOLOAD	227
of SYSCTLGS	210
Optional Cluster Data Report at Restore Time	217
Parameters for Non-VSAM Data Sets	227
PASSWORD, CATALOG	224
PASSWORD, MASTER	224
PDS Member Restore	211
Physical Alternate Indexes	219
Process the Queue, Example	430
RESTORE Command Parameters for VSAM Clusters	228
Restore Considerations for GDG Data Sets	212
Restore Volume Selection	209
Sequencing Restore Requests	208
Space Allocation	211
Tape Pull List for Queued Requests	240
to Predefined and Reusable Clusters	217
Restore Command	
Password	224
Restore Command Parameters	
for Non-VSAM Without DEFER	227
for VSAM Data Components, Without Defer	234
for VSAM Index Components, Without Defer	237
Honored Without DEFER	227
RESTORE Command Parameters	
for VSAM AIXS Without Defer	229
for VSAM Clusters, Without Defer	230
for VSAM Without Defer	228
RESTORE/RECOVER	207
and Compression Products	15
Files Data Set Updates for	222
Restrictions	222
Sample Report	249
Restored Data Sets	
Exempting	168
Restrictions	
ARCHIVE	201
DARCHIVE	411
General	20
in Restore and Recover of VSAM Clusters	213
Incremental Backup	184
Move/Copy	334
REPORT	101
RESTORE/RECOVER	222
Sequential Migration to Tape	377
TSO	421
VSAM Backup and Archive Restrictions	174

Retention Control	165
RETPD	186, 191, 203
Reusable & Predefined Clusters, Restoring to	217
REVAIL/XB-SPOOL	15
ROLLOFF	59
RPT Parameter	260
RPT=CIA	187, 192
Rules	
DSCL SCAN and FIND	28
DSCL SELECT and EXCLUDE	31
Runs	
Monday Through Saturday	445
Sunday, Incremental BACKUP	445
S	
SCAN	
Command, Implicit	362
Command, Implicit Migration to Tape	379
DSCL Rules	28
Parameters, Sequential Migration	373
SCONSPAC	56
SCRATCH	204
Scratch/Catalog Defaults	332
SECFLAGS	64
Secondary allocation amount (number)	56
Secondary Extent Consolidation	214
Security	
Considerations for VRECOVER	257
Considerations, VBACKUP	254
PASSTYPE, Password protection type	62
Password Detail Group (P)	114
PASSWORD, Password-protected (yes/no)	62
Password-Protected Data Sets	16
RACFIND, RACF-indicated (yes/no)	64
SELECT	
CRITERIA Field Name Descriptions	50
DSCL Rules	31
Logic	68
Logic, FMS	95
Parameters, Sequential Migration	373
Selection	
Criteria, Data Sets, Sequential Migration	373
Restore Volume Selection	209
Volume, for Alternate Indexes	219
Volume, for Multivolume Clusters	220
Separator Pages	10
Sequencing	
ARCHIVE Report Options	170

Archive Report Sequencing Options	200
Sequencing Restore Requests	208
Sequential Migrate to Tape	
Examples	453
Sequential Migration to Tape	371
Catalog Concerns	382
Concerns	382
Condition Codes	377
General Information	372
Restrictions	377
Tape First File, Creation of	375
Versus Archival	374
Settings of Condition Codes	297
SIMULATE	202 - 203
SIMULATE Command	35, 81
SMF Report, Sample JCL for executing	138
Space Allocation	
Restore	211
SPACOPTN	55 - 56
Specifications	
Reblocking	339
Specifying	
Merge Output Options	315
What to Merge	314
Speed Option	214
STATUS	417
Storage Requirements	11
Support	
Alternate Index	175
SVCMODE, SVC mode field for VTOC statistics	65
SYNTAX	
ARCHIVE	191
BACKUP	186
DSCL CRITERIA Parameter	47
DSCL EXCLUDE Command	70
DSCL FIND Command	42
DSCL SCAN Command	38
DSCL SELECT Command	44
EXCLUDE Command - FMS PROC	96
PASSWORD Command - FMS PROC	83
RECOVER Command - FMS PROC	245
SCAN Command - FMS PROC	84
SELECT Command - FMS PROC	85
SET Command - FMS PROC	81
SYSTLG	
Archival of	201
Backup of	171
Parameter	227
Processing	337

Restore of	210
Sysparms	
UNMOVEAR	202
DASDPOOL	209
DSBCATSZ	404 - 405
DSCBLMOD	51, 54
DSCBLUSD	52, 54
DSCBSVMD	65
DSCLMSGs	26, 67
DSNSELRE	44
FLYCOUNT	10
Idle Space Release	355
IXUSRNOT	296
PASSNEWN	16
PRIALLOC	17
RETRETPD	39
RPTCLSEL	102
SECURVOL	257
SELECTRE	44
SYSPARMO	456 - 457
UNMOVEAR	184
VOLSELRE	46
VSCATPTN	39
VSCIACAT	116
VSCIADET	116
VSCOMPNM	112
VSCRIMAX	47
VSUSEBLK	105 - 106
System temporary non-VSAM data set (yes/no)	65

T

TABLE=	86
Tailoring to User Needs	8
Tape	
Archive, Making Additional Copies of	310
Data Sets on, Expiration Dates Assigned to	374
Duplicate, Recatalog to, JCL	383
First File, Creation of, Sequential Migration	375
Migration, Duplicate Copies of	382
Processing, Multivolume	374
Pull List Condition Codes	240
Pull List for Queued Restore Requests	240
SCAN Command, Implicit Migration to	379
Sequential Migration to Tape, JCL	378
TAPEPULL Processing	317
Tape Management Systems	
Compatibility with Sequential Migration	375
Target Allocation	332

Task	
Performance	169
Task Restart Considerations	170
TDSN=	86
TEMP, System temporary non-VSAM data set	65
TIMEDATE Parameter	307
TODAY	94
Track Image Processing	
for Volume-Level Functions	13
Track Overflow Data Sets	253
TRKS, Tracks allocated	65
TSO Command Processor	409
DERASE Command, RESTORE Parameter	420
General Information	410
Restrictions	421
User Exits	422
Type of DSNINDEX record	90

U

UNCATALOG	204
Unexpired Data Sets	
Delete	173, 215
UNIT Parameter	305
Unmovables	201
UNMOVE, Unmovable attribute (yes/no)	66
Updates	
Index, JCL	306
to Files Data Set, for RESTORE/RECOVER	222
USEDT	94
USEDT, Last Use date	52
User Exits	
Copy command	341
Move command	341
TSO	422
User Field Information in DSNINDEX record	94
USERFLD	94
Utility	
Archive Index Rebuild Utility	304
IXMAINT	296

V

Validation of Data Set Name	7
VBACKUP	
Archive Management Considerations	254
Commands and Parameters, DSCL	255
Function	253
Security Considerations	254

VBACKUP/VRECOVER	251
VCCOPYEX	341
VCDMIGEX	341
Verify of VSAM Clusters by SAMS:Disk	172
VERSION	94
Version # of ARCHIVE DSNINDEX records	95
Version # of BACKUP DSNINDEX records	95
Version # of the DSNINDEX record	94
VERSIONA	95
VERSIONB	95
VOLCNT, Data set allocated volume count	66
VOLDELETE	
Command Parameters Description	301
VOLSEQNO, Volume sequence number	66
Volume	203
Defragmentation	248
Defragmentation, and Implicit Recovery	242
Defragmentation, Example	434
Delete Expired	299
Index Command, BUILDARC	308
Prefix	8
Reset COPY Parameter	307
Restore Volume Selection	209
Selection for Multivolume Clusters	220
Selection, for Alternate Indexes	219
Volume and Data Set Index Updates (IXUPDATE)	306
VTOCS, Backup of	169
Volume-Level	
BACKUP Examples	432
Functions, Track Image Processing for	13
Overview of Processing	252
Processing, VBACKUP/ VRECOVER	252
RECOVER Examples	433
VOLUMES=	86
VRECOVER	
Commands and Parameters	258
Function	257
Security Considerations	257
VREPORT	
ALLOCS, Volume Allocation Summary	107
ATTRBS, Volume Attribute Summary	108
DISTRs, Volume Distribution Summary	108
Examples	425
LAYOUT, Volume Layout	108
VREPORT, FREESP, Free Space Detail	108
VSAM	
BACKUP and ARCHIVE Considerations	172
Backup and Archive Restrictions	174
CASPLITS, Control Area splits (number)	57

Clusters, Processing Considerations	348
Clusters, Restore & Recover Restrictions	213
Clusters, RESTORE for, Without Defer	230
DSCL PASSWORD Command	36
Error Recovery	349
EXTENTS in the data set (number)	58
FMS PASSWORD Command	83
GDGSTAT, Generation data group status	59
KBYTES, Kilobytes allocated to the data set	60
PCTUSED, Percent used (number)	62
Recataloging for VSAM Auto-Restore	174
Restore and Recover of VSAM Clusters	213 - 224
RESTORE Command Parameters for VSAM Clusters	228
SAMS:Disk Verify of VSAM Clusters	172
VSAMORG, VSAM organization type	67
VSAM Clusters	
Access Methods for	174
VSAM Data Components	
RESTORE for, Without Defer	234
VSAM Index Components	
RESTORE for, Without Defer	237
VSAMORG, VSAM organization type	67
VSUSEBLK, BLKSZ, Block size (number)	56
VTOC	
Backup of Volume	169
VTOCIX	46, 71
VTOCMSG=	81
VTOCS	86
VVDS	46, 72
Bypassing overhead	107
X	
XCOPY	263

